

UNIVERSITÉ DE MONTRÉAL

SYSTÈME DE FORMATION D'OPÉRATEURS DE RÉSEAUX
PAR RÉALITÉ VIRTUELLE

ALAIN DAIGLE
DÉPARTEMENT DE GÉNIE ÉLECTRIQUE
ET DE GÉNIE INFORMATIQUE
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION
DU DIPLÔME DE MAÎTRISE ÈS SCIENCES APPLIQUÉES
(GÉNIE ÉLECTRIQUE)

JANVIER 1998



National Library
of Canada

Acquisitions and
Bibliographic Services

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque nationale
du Canada

Acquisitions et
services bibliographiques

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence

Our file Notre référence

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-38671-6

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Ce mémoire intitulé:

SYSTÈME DE FORMATION D'OPÉRATEURS DE RÉSEAUX
PAR RÉALITÉ VIRTUELLE

présenté par: DAIGLE Alain

en vue de l'obtention du diplôme de: Maîtrise ès sciences appliquées

a été dûment accepté par le jury d'examen constitué de:

M. OLIVIER Guy, Ph.D., président

M. MARCEAU Richard J., Ph.D., membre et directeur de recherche

M. GRANGER Louis, M.Sc., membre

Remerciements

De nombreuses personnes ont contribué à la réalisation de ce travail de maîtrise, qui fait partie du projet ESOPE-RV. Ces personnes, de par leurs encouragements, leurs conseils et leur assistance, ont fait de ce projet une réalité. Je tiens donc à remercier premièrement M. Louis Marquis et M. Robert Gauthier, de la division Automatisation et support à la formation d'Hydro-Québec, pour avoir imaginé le projet innovateur qu'a été ESOPE-RV ainsi que pour la confiance qu'ils ont su démontrer en partageant sa réalisation avec 5 étudiants de l'École Polytechnique et de l'université McGill.

Je tiens également à remercier M. Richard J. Marceau, mon directeur de recherche, de même que M. Alfred Malowany, directeur de recherche pour les étudiants de McGill, pour leurs encouragements, leur soutien et leurs conseils. Je remercie également les autres membres de ce projet, soit Patrick Desbiens, Etienne Garant, Alex Okapuu von-Vhe, Amir Shaikh ainsi que Jean-Claude Rizzi, dont les efforts et les longues heures de travail ont contribué à faire de ce projet une réalité.

Finalement, je tiens à remercier mon épouse Stéphanie, ainsi que les membres de ma famille, pour leur patience et leurs nombreux encouragements pendant la réalisation de ce projet et lors de la rédaction de ce mémoire.

Résumé

Le projet ESOPE-RV vise de façon générale l'élaboration d'un logiciel employant la réalité virtuelle en vue d'améliorer la formation des opérateurs de postes. La qualité de la formation de ces opérateurs de poste est capitale étant donné la nature dangereuse de leur travail. Le projet ESOPE-RV permet donc à l'opérateur en formation d'apprendre à réaliser des plans de manœuvres dans un environnement 3D représentant un poste typique, tout en respectant les règles de sécurité et d'exploitation d'Hydro-Québec. Ce logiciel intègre de nombreuses technologies innovatrices : la réalité virtuelle, la reconnaissance de la voix, l'aide multimédia, la conversion automatique en 3D de schémas unifilaires de poste ainsi qu'un système-expert de validation de plans de manœuvres développé par Hydro-Québec.

Le projet, dont l'architecture de type distribué est constituée d'une multitude de processus répartis sur plusieurs ordinateurs, utilise un algorithme de conversion en 3D du schéma unifilaire basé sur les six topologies courantes de postes. La représentation du poste électrique en 3D est réalisée à l'aide de WorldToolkit qui donne accès à une librairie de fonctions C. Ces fonctions permettent de créer l'environnement virtuel et d'effectuer la gestion des interactions de l'opérateur qui se font par le biais de la souris et de la commande vocale. Les manœuvres effectuées à partir d'ESOPE-RV sont validées par le système-expert Opérateur+ d'Hydro-Québec. L'intégration de ce système-expert qui assiste les opérateurs dans leur formation se fait par l'entremise

d'un processus appelé esope qui gère et valide les communications dirigés vers le système-expert.

La qualité principale d'ESOPE-RV est de pouvoir apprendre à l'opérateur à effectuer des plans de manœuvres simples mais réalistes dans un environnement virtuel où les erreurs de manipulations demeurent sans risque. De plus, l'apprentissage y est facilité grâce à une aide interactive aisément accessible permettant de guider l'opérateur. La conception distribuée d'ESOPE-RV facilite grandement la mise à jour d'un aspect du logiciel tout en minimisant les effets négatifs sur l'ensemble des modules.

Abstract

The purpose of the ESOPE-VR project was to upgrade a software-based training system to integrate a virtual reality interface in order to increase the quality of training of station operators. Because of the inherent dangers attached to the specific type of work performed by station operators, the quality of their training is capital. The ESOPE-VR project allows the operators in training to learn manual switching operations in a 3D environment representing a typical station while respecting safety and operating rules of the Hydro-Québec system. This immersive training system therefore integrates numerous innovative technologies such as virtual reality, speech recognition, multimedia support, automated conversion of single-line diagrams to three-dimensional models and a Hydro-Quebec knowledge-based expert system that validates the operators' switching operations.

The ESOPE-VR project is based on a distributed architecture and consists of a multitude of processes divided on a number of computers. The project uses a single-line diagram-to-three-dimension conversion algorithm to generate the station environment used for training. This 3D representation of the station is made with the help of WorldToolkit that allows access to a library of C functions. These functions are used to create the virtual environment and to manage the mouse and speech recognition input given by the operator. The switching operations carried out with ESOPE-VR are validated by the Hydro-Quebec knowledge-based expert system

OPERATOR+. The integration of OPERATOR+ to ESOPE-VR is made possible by a process called esope that manages and validates all communications between the two.

The principal quality of ESOPE-VR is its ability to help the station operator execute simple but realistic manual switching operations in a virtual environment where errors can be performed without risk. The easily accessible interactive support offered by the system facilitates the operator's training by guiding him throughout the switching operations. The distributed design of Esope-VR facilitates the updating of a project's aspect while minimizing negative effects on other processes.

TABLE DES MATIÈRES

REMERCIEMENTS.....	IV
RÉSUMÉ.....	V
ABSTRACT	VII
LISTE DES TABLEAUX	XII
LISTE DES FIGURES.....	XIII
LISTE DES ABRÉVIATIONS.....	XIV
INTRODUCTION	I
TRAVAIL DES OPÉRATEURS	1
FORMATION DES OPÉRATEURS	5
LACUNES DE LA FORMATION ACTUELLE	7
OBJECTIF DE CETTE RECHERCHE	8
LA RÉALITÉ VIRTUELLE	9
PERTINENCE DE CETTE RECHERCHE.....	12
CONTRIBUTION PERSONNELLE À CETTE RECHERCHE	13
CONCLUSION	15
2. DESCRIPTION DU SIMULATEUR ESOPE-RV	16
2.1 INTRODUCTION	16
2.2 REPRÉSENTATION 3D DE L'INSTALLATION	16
2.2.1 <i>Choix d'un outil de création d'environnement virtuel</i>	16
2.2.2 <i>WorldToolKit de Sense8</i>	19
2.2.3 <i>Création et gestion de l'environnement 3D</i>	19
2.3 LOGICIEL OPÉRATEUR+	26
2.4 CONVERSION DE SCHÉMAS UNIFILAIRES	28
2.5 COMMANDE VOCALE	30
2.6 ENVIRONNEMENT SONORE	31

	x
2.7 SUPPORT MULTIMÉDIA.....	32
2.8 PÉRIPHÉRIQUES D'ENTRÉE.....	33
2.9 CONFIGURATION ET DÉMARRAGE.....	35
2.10 CONCLUSION.....	40
3. ARCHITECTURE D'ESOPE-RV.....	41
3.1 INTRODUCTION.....	41
3.2 DESCRIPTION DES ÉLÉMENTS PHYSIQUES.....	41
3.3 MODULES ET PROCESSUS DE L'ARCHITECTURE.....	43
3.4 COMMUNICATIONS ENTRE PROCESSUS.....	49
3.5 CONCLUSION.....	56
4. INTÉGRATION D'OPÉRATEUR+.....	57
4.1 INTRODUCTION.....	57
4.2 EXIGENCES D'ESOPE-RV.....	57
4.3 CONTRAINTES DE RÉALISATION.....	60
4.4 PROCESSUS D'INTERFACE ESOPE.....	61
4.5 MODIFICATIONS AU VALIDEUR DE MANŒUVRES.....	63
4.6 MOYENS DE COMMUNICATION.....	66
4.7 FORMAT DES MESSAGES.....	67
4.7.1 Messages de type « manœuvre ».....	69
4.7.2 Messages de type « interrogation ».....	71
4.7.3 Messages de type « navigation ».....	79
4.7.4 Messages de type « erreur ».....	81
4.8 GESTION DES MESSAGES PAR ESOPE.....	82
4.8.1 Lecture des messages destinés au valideur.....	86
4.8.2 Validation du message.....	87
4.8.3 Conversion des messages.....	88
4.8.3.1 Messages de type « manœuvre ».....	89
4.8.3.2 Messages de type « navigation ».....	92
4.8.3.3 Messages de type « interrogation ».....	92
4.8.3.4 Communs à tous les types de message.....	94
4.8.4 Envoi de la commande au valideur.....	95
4.8.5 Lecture des résultats provenant du valideur.....	96
4.8.6 Validation du résultat.....	96
4.8.7 Conversion et envoi du résultat.....	101

4.9	CONCLUSION	104
5.	PERFORMANCES D'ESPE-RV	105
5.1	INTRODUCTION	105
5.2	DÉMARRAGE DU SIMULATEUR	105
5.3	CRÉATION DE L'INSTALLATION EN RÉALITÉ VIRTUELLE	106
5.4	RÉALISATION DU PLAN DE MANŒUVRES TYPE	108
5.5	CONCLUSION	113
6.	LIMITATIONS.....	114
6.1	INTRODUCTION	114
6.2	LIMITATIONS DE TYPE MATÉRIEL	114
6.3	LIMITATIONS DE TYPE LOGICIEL	116
6.4	CONCLUSION	119
7.	CONCLUSION	120
	BIBLIOGRAPHIE	127
	ANNEXES.....	138
1.	PLAN DE MANŒUVRE TYPE	139
2.	SCHÉMA UNIFILAIRE TYPE	140
3.	STRUCTURES DE DONNÉES UTILISÉES	141

Liste des tableaux

<i>Tableau 2-1 Comparaison des outils.....</i>	<i>18</i>
<i>Tableau 4-1 Messages de type « manœuvre ».....</i>	<i>70</i>
<i>Tableau 4-2 Messages de type « interrogation »</i>	<i>72</i>
<i>Tableau 4-3 Messages de type « navigation ».....</i>	<i>80</i>
<i>Tableau 4-4 Messages de type « erreur »</i>	<i>82</i>
<i>Tableau 4-5 Messages d'erreur du valideur de manœuvres.....</i>	<i>98</i>
<i>Tableau 4-6 Erreurs de manœuvres possibles.....</i>	<i>100</i>
<i>Tableau 4-7 Résultats possible d'une interrogation.....</i>	<i>102</i>

Liste des figures

<i>Figure 2-1 Panneau de la salle de commande</i>	<i>21</i>
<i>Figure 2-2 Extérieur du poste</i>	<i>22</i>
<i>Figure 2-3 Boutons du panneau de commande.....</i>	<i>24</i>
<i>Figure 2-4 Manette du panneau de commande.....</i>	<i>24</i>
<i>Figure 2-5 Main virtuelle devant le panneau de commande.....</i>	<i>25</i>
<i>Figure 2-6 Le valideur de manœuvres.....</i>	<i>27</i>
<i>Figure 2-7 Extrait d'une séquence vidéo sur l'ouverture d'un sectionneur à perche</i>	<i>32</i>
<i>Figure 2-8 Déplacement avec la souris 2D.....</i>	<i>33</i>
<i>Figure 2-9 Première partie du fichier de configuration.....</i>	<i>35</i>
<i>Figure 2-10 Seconde partie du fichier de configuration</i>	<i>36</i>
<i>Figure 2-11 Messages affichés lors du démarrage</i>	<i>39</i>
<i>Figure 3-1 Modules dans le simulateur ESOPE-RV</i>	<i>45</i>
<i>Figure 3-2 Processus du simulateur ESOPE-RV</i>	<i>49</i>
<i>Figure 3-3 Architecture complète du simulateur 3D.....</i>	<i>55</i>
<i>Figure 4-1 Processus esope</i>	<i>62</i>
<i>Figure 4-2 Gestion des messages par esope</i>	<i>85</i>
<i>Figure 4-3 Traitement des erreurs de manœuvre.....</i>	<i>99</i>
<i>Figure 4-4 Messages conditionnels.....</i>	<i>103</i>

Liste des abréviations

WTK	Processus worldtoolkit
ESO	Processus esope
ERV	Processus esope-rv
A_V	Processus audiovidéo
VBX	Processus verbex
SCADA	System Control And Data Acquisition, appareil permettant l'acquisition de données de même que la commande de systèmes à distance.
ASCII	American Standard Code for Information Interchange, code définissant une table de caractères utilisée par tous les ordinateurs.
TCP/IP	Transmission control protocol/internet protocol (protocol de communication utilisé par l'Internet).
SGI	Silicon Graphics inc. (un fabricant d'ordinateurs spécialisés dans les applications graphiques).
PC	Personnal computer (un ordinateur dit personnel, utilisant habituellement un processeur compatible X86).
R4000	Microprocesseur fabriqué par MIPS
R3000	Microprocesseur fabriqué par MIPS
MIPS	Filiale de SGI qui fabrique des microprocesseurs
ELAN	Modèle d'ordinateur SGI
3D	3 Dimensions, qui possède trois dimensions
CER	Centre d'Exploitation Régional, centre de contrôle d'Hydro-Québec
HMD	Head-mounted display, casque qui permet de projeter des images stéréoscopiques devant les yeux.
MIT	Massachusset Institute of Technology, université prestigieuse située près de Boston.
NASA	National Aeronautics and Space Agency, organisme américain responsable de l'exploration de l'espace.
AIFF	Apple Image File Format, format de fichier de sons créé par Apple computers
UNIX	Système d'exploitation créé par AT&T.

Introduction

Ce chapitre présente la problématique sur laquelle la recherche est basée, soit la formation des opérateurs de postes et de centrales. Le travail d'opérateur, de même que le type de formation que l'opérateur reçoit, y sont décrits. La solution proposée, visant à combler les lacunes de la formation actuelle, est ensuite présentée sommairement.

Travail des opérateurs

Les opérateurs de postes et de centrales d'Hydro-Québec ont comme tâche principale la manœuvre des équipements électriques du site. Ces manœuvres, qui visent essentiellement à ouvrir et à fermer des disjoncteurs et des sectionneurs, peuvent être prévues d'avance dans le cadre d'un programme routinier d'entretien et sont alors préalablement approuvées par le centre d'exploitation régional (CER), endroit où le réseau électrique est géré globalement à l'intérieur d'une région administrative d'Hydro-Québec. Toutefois, lors de situations d'urgence, suite à des conditions climatiques particulières ou à des bris d'équipements, les manœuvres peuvent être plus spontanées, ce qui entraîne parfois des erreurs de manœuvres. Les manœuvres préalablement approuvées par le CER peuvent tout de même entraîner des conséquences fâcheuses pour le réseau, si l'opérateur commet une erreur.

Le transport et la distribution de l'électricité sont des opérations qui comportent certains dangers. En effet, l'opérateur peut à tout moment faire une action qui aura des conséquences néfastes pour lui ou pour l'équipement. Dans sa formation, on lui apprend ce qu'il peut faire et ce qu'il ne peut pas faire dans une installation. Par exemple, il ne peut pas toucher à une barre sous tension avec sa main mais il peut le faire avec la perche, l'appareil utilisé pour détecter la présence de tension. Cependant, certaines conditions spécifiques peuvent rendre dangereuse une manœuvre habituellement sécuritaire. Par exemple, une journée pluvieuse ou très humide peut rendre dangereux l'utilisation de la perche. Une distraction peut aussi amener l'opérateur à commettre une erreur désastreuse, tel le contact de la perche avec une barre dont la tension dépasse la capacité maximale de la perche. Les périmètres de sécurité des appareils, qui délimitent un espace interdit à l'opérateur autour des appareils, doivent aussi être respectés en tout temps. Si l'opérateur circule à l'intérieur de ce périmètre, le risque de s'infliger des blessures suite à une décharge électrique augmente. Ainsi, ces erreurs peuvent se traduire par l'électrocution de l'opérateur et/ou un bris de l'équipement et/ou une coupure de service aux clients.

Pendant l'exécution de manœuvres dans un poste, l'opérateur doit donc toujours demeurer vigilant. Une manœuvre peut entraîner le dépassement des limites sécuritaires d'utilisation des appareils électriques et causer des dommages sérieux. Par exemple, une tension trop élevée sur un appareil peut entraîner un claquage. Dans cette situation, un arc électrique se forme et peut souvent provoquer l'explosion des

isolateurs en porcelaine. Les éclats de porcelaine projetés sont aussi coupants que des éclats de verre. Une personne en périphérie de l'équipement risque alors d'être électrocutée par l'arc et d'être gravement blessée par les éclats de porcelaine. Un courant trop élevé à l'intérieur d'un appareil peut aussi conduire à des conséquences néfastes, telle une explosion.

Un autre danger potentiel provient de la difficulté d'identification des appareils et d'orientation dans un poste. Cette difficulté s'explique par plusieurs facteurs. Premièrement, une partie de l'équipement est disposée sur le sol alors qu'une autre partie est fixée à des structures d'acier, à plusieurs mètres du sol. De plus, certains postes ont des dimensions imposantes: ils sont grands comme plusieurs terrains de football et contiennent plusieurs centaines d'appareils. Il est alors plus difficile de trouver et de mémoriser l'emplacement des appareils. L'emploi des schémas unifilaires est également un facteur qui vient compliquer le travail de l'opérateur. Ces schémas, de par leur degré d'abstraction, ne représentent pas l'ensemble des appareils dans le poste. En effet, une seule phase du système triphasé est représentée et les symboles utilisés pour désigner les appareils ne reflètent pas la dimension réelle de ces derniers. En outre, le schéma unifilaire ne permet pas de savoir si un appareil est au sol ou fixé à une structure d'acier. Finalement, il arrive que les appareils soient très près les uns des autres à cause de contraintes physiques. Dans ces situations, les risques d'une mauvaise identification d'un appareil sont augmentés. Dans le cas des structures,

un autre facteur affectant l'identification des appareils est la hauteur des appareils. Les appareils sont alors difficiles à identifier visuellement.

Les postes étant souvent dans des zones urbaines, l'opérateur n'est pas le seul qui court le risque de se blesser. Par exemple, la vie de travailleurs ou de voisins peut être menacée par les actions de l'opérateur.

Les risques de blessures ou de mort accidentelle sont donc très présents dans le travail d'un opérateur. Heureusement, ce genre d'accident est plutôt rare. En effet, une étude faite auprès d'entreprises de distribution et de transport d'électricité américaines indique que 1.25% des erreurs de manœuvre causent des blessures (Beare, 1997). Les bris d'équipement sont cependant plus fréquents, tout comme les coupures accidentelles d'alimentation. Cette même étude indique que 3.5% des erreurs de manœuvres causent des bris d'équipement et que les coupures accidentelles d'alimentation représentent 51.1% des erreurs de manœuvres. Ces erreurs engendrent des pertes ou des coûts tant à l'employeur qu'à ses clients et peuvent ternir la réputation de ce dernier face à ses clients. Par conséquent, la qualité de la formation reçue par les opérateurs est, dans ce contexte, très importante.

Formation des opérateurs

Les opérateurs de poste, qui doivent manipuler les équipements électriques dans les postes, bénéficient d'une période de formation peu après leur embauche. Cette formation débute par 30 jours de cours en classe suivi par 5 mois de formation sur le lieu de travail. La scolarité minimale pour être opérateur chez Hydro-Québec est un diplôme d'études secondaires (DES). L'objectif de la formation est de faire des manœuvres sécuritaires, sans danger pour l'équipement et de maintenir le service à la clientèle (Hydro-Québec, 1993).

Hydro-Québec possède un centre de formation pour ses opérateurs de postes et de centrales. Ce centre a pour but l'apprentissage des règles de base lors de la manipulation d'équipements électriques. On y enseigne aussi la séquence des principales manœuvres normalement réalisées dans un poste.

Le matériel pédagogique est surtout sous la forme de documentation écrite. On y retrouve les règles d'exploitation d'Hydro-Québec, la description des pièces d'équipement d'un poste et leurs fonctions ainsi que des schémas unifilaires de postes existants. On y enseigne également des plans de manœuvres typiques et/ou réels. De plus, on retrouve des photos des appareils ainsi que des vidéos montrant certains éléments d'un plan de manœuvre. Lorsque possible, on effectue la visite de postes dans la périphérie du centre de formation pour observer/réaliser des manœuvres.

Le centre de formation possède aussi une salle de contrôle miniature permettant de se familiariser avec les panneaux que l'on retrouve dans les postes. On peut manipuler les appareils se trouvant sur ce panneau en actionnant le bouton ou la manette appropriée. Comme un simulateur analogique est intégré au panneau de contrôle et est connecté aux différents boutons et manettes, on peut aussi consulter les cadrans pour connaître la topologie de ce réseau imaginaire de même que les différentes valeurs de courant, de tension et de puissance.

Depuis quelques années, le centre de formation emploie également un logiciel de validation de manœuvres, le valideur de manœuvres d'OPÉRATEUR+. À l'aide de ce logiciel, l'opérateur en formation peut exécuter les manœuvres normalement requises pour effectuer un certain travail. Ce logiciel, qui contient un engin de validation de manœuvres ainsi qu'un engin de calcul d'écoulement de puissance, facilite l'apprentissage des séquences d'opérations qu'un opérateur doit réaliser dans son travail. L'engin de validation de manœuvres s'assure que l'opérateur respecte les règles d'exploitation de l'entreprise, alors que l'engin de calcul d'écoulement de puissance confirme que les limites de tension et de courant des appareils ne sont pas dépassées suite à une manœuvre de l'opérateur. Lorsque les règles d'exploitation ou les limites des appareils ne sont pas respectées, un message s'affiche à l'écran indiquant à l'opérateur la nature de son erreur.

Lacunes de la formation actuelle

Ces moyens de formation présentent cependant certaines lacunes: on apprend les plans de manœuvres dans une salle de cours plutôt que dans la cour aux équipements d'un poste. Il peut parfois y avoir un problème de transfert de connaissances lorsque vient le temps d'effectuer ce plan de manœuvres dans un poste réel. De plus, à cause de la grande panoplie d'appareils en service dans nos postes, du respect approximatif de certaines normes dans la construction des postes, de considérations physiques et économiques, on enseigne habituellement les situations les plus fréquentes. Cette diversité peut amener l'opérateur à agir de façon erronée face à un problème qu'il n'a jamais rencontré en classe. Une étude récente d'Hydro-Québec indique qu'environ 17% des erreurs de manœuvres sont causées par une formation déficiente (David, 1996). La principale lacune de la formation actuelle selon cette étude est l'absence de formation adéquate directement sur le poste où l'opérateur travaille.

La salle de contrôle miniature permet de reproduire un nombre assez limité de situations. On n'y retrouve que les équipements les plus fréquents dans les postes. Parce qu'il est impossible de représenter un poste existant d'Hydro-Québec, le rôle de cette salle est donc avant tout de permettre l'apprentissage de notions d'électricité.

Les moyens actuels de formation des opérateurs permettent à ces derniers de bien saisir la séquence des opérations qu'ils doivent réaliser pour arriver à un but

particulier. Cependant, il arrive que, lorsqu'il est temps d'appliquer une opération particulière, les opérateurs effectuent l'opération sur le mauvais équipement. Ce genre d'erreur peut avoir des effets fâcheux comme endommager l'équipement, causer des blessures ou couper l'alimentation à certains clients. Un rapprochement entre l'environnement de formation et le milieu de travail semble donc nécessaire pour améliorer la compétence des opérateurs.

La révolution informatique des 15 dernières années a donné naissance à plusieurs technologies qui, combinés aux moyens actuels de formation des opérateurs, permettent d'envisager un renouvellement de la façon de former les opérateurs dans le but d'accroître la qualité de l'enseignement reçu.

Objectif de cette recherche

Étant donné la problématique décrite précédemment, la question à laquelle ce projet tente de répondre est la suivante: « *Comment rapprocher le milieu de formation des opérateurs de leur environnement de travail ?* ». Le moyen préconisé est l'utilisation de la réalité virtuelle pour reproduire le plus exactement possible l'environnement de travail de l'opérateur dans le cadre de sa formation. Les possibilités offertes par la réalité virtuelle sont immenses, en particulier pour la formation de personnes. L'aspect le plus intéressant est sans doute l'immersion dans le milieu de travail. Il est en effet possible de reproduire avec un niveau de détails saisissant l'environnement de travail

de la personne en formation. Cette personne a alors la possibilité de se déplacer dans ce monde virtuel. En plus de la réalité virtuelle, plusieurs technologies et techniques sont aussi utilisées pour simuler le plus fidèlement l'environnement d'un poste électrique. La formation peut alors s'effectuer sur une représentation presque identique de postes existants et sur des équipements semblables. Ce nouvel outil de formation permet ainsi d'éliminer, chez les opérateurs, le problème de transfert de connaissances de l'environnement de formation vers l'environnement de travail. L'objectif général de la présente recherche est donc de *concevoir et de réaliser un simulateur de postes utilisant la réalité virtuelle*. Ce simulateur est appelé « **ESOPE-RV** ».

La réalité virtuelle

La réalité virtuelle est une façon de représenter un environnement réel le plus fidèlement possible ou un environnement imaginaire de façon synthétique, à l'aide d'un ordinateur (Kalawsky, 1993). Cette représentation est visuelle et idéalement, auditive et sensorielle (Kalawsky, 1993).

Lorsqu'elle est utilisée pour la formation d'opérateurs, la réalité virtuelle permet de représenter, à l'aide d'un ordinateur puissant, presque n'importe quel poste existant. La représentation visuelle du poste peut se faire de plusieurs façons. La solution la plus simple et la moins coûteuse est l'utilisation d'un écran d'ordinateur pour présenter l'environnement virtuel. Cependant, la sensation d'immersion dans l'environnement

est faible. Une autre approche plus réaliste mais moins abordable est le casque à écrans intégrés (*head-mounted display*). L'environnement virtuel est dans ce cas représenté sur deux petits écrans à cristaux liquides placés à l'intérieur du casque, devant les yeux. Ce casque est habituellement combiné à un système de positionnement de la tête qui permet de changer la représentation de l'environnement en fonction des mouvements et de l'orientation de la tête de l'utilisateur. La sensation d'immersion est dans ce cas beaucoup plus élevée.

La reproduction de l'environnement sonore est également importante afin de rendre la simulation réaliste. Les sons dans la réalité permettent de mieux se situer et de s'orienter. Ils peuvent également indiquer qu'un événement s'est produit (Garant, Daigle et al., Mai 1995). Le matériel nécessaire pour reproduire efficacement des sons est plus accessible et comprend essentiellement un amplificateur ainsi que des haut-parleurs de qualité. Les sons peuvent être des sons synthétiques ou des enregistrements de sons réels.

Finalement, l'aspect sensoriel facilite la manipulation des objets 3D de l'univers virtuel en donnant une contre-réaction aux mouvements de l'utilisateur. Cette contre-réaction aide à doser les forces appliquées par l'utilisateur aux objets de l'environnement (Garant, Daigle et al., Mai 1995). Il existe des périphériques d'entrée qui permettent à l'utilisateur de sentir une contre-réaction à ces mouvements. Ces périphériques en sont cependant à leur début et n'ont pu être utilisés dans le cadre de ce projet.

L'emploi du terme « réalité virtuelle » est relativement récent, d'une dizaine d'années environ. Cependant, les premières expériences dans ce domaine datent de 1956, alors que Morton Heilig a conçu le « Sensorama » (Kalawsky, 1993). Cet appareil permettait de simuler une balade en moto dans les rues de Manhattan en combinant des images 3D, des sons en stéréo, des vibrations, des odeurs et même des vents. Le boom dans le domaine de la réalité virtuelle a cependant commencé vers le milieu des années 80 lorsque le *Massachusetts Institute of Technology* (MIT) et la *National Aeronautics and Space Agency* (NASA) ont mis sur pied des laboratoires de réalité virtuelle. De nos jours, il existe plusieurs projets de formation qui utilisent la réalité virtuelle. Ces projets touchent à presque tous les domaines. Voici quelques projets de formation réalisés ces dernières années:

- Formation du personnel de la National Guard qui effectue la maintenance des chars d'assaut M1A1 Abrams, des véhicules de combat M2A2 Bradley et des systèmes de missiles TOW II (McLin, 1995);
- Formation des travailleurs d'une chaîne de montage à une usine hautement robotisée de Motorola à Boynton Beach, Floride, USA (Wittenberg, 1995);
- Simulation d'un poste électrique sous-terrain pour la formation du personnel de maintenance, (Mizukami, 1995);
- Formation des astronautes de la NASA pour la mission d'entretien du télescope spatial Hubble (Loftin, 1995);

- Formation du personnel militaire en charge du triage des blessés dans des situations de crise (Dumay, 1995).

La médecine (chirurgie, anesthésie, etc.) est aussi un domaine d'application fréquent de la réalité virtuelle à des fins de formation (Hunter, 1993).

Les effets physiologiques de la réalité virtuelle pourraient cependant freiner son utilisation pour la formation. En effet, selon Stephen Strauss (1995), près de 30% des utilisateurs de simulateur en réalité virtuelle souffrent de sueurs froides, de nausées, de maux de tête ou de vomissements. De plus, une étude de l'Université d'Edimbourg réalisée en 1993 concluait que les casques à écrans intégrés (*head-mounted display*) utilisés fréquemment avec des simulations en réalité virtuelle pouvaient causer des dommages cérébraux permanent (Johnson, 1995). Le simulateur ESOPE-RV n'emploie pas ce type de casque.

Pertinence de cette recherche

En permettant à l'opérateur de s'entraîner dans un environnement très semblable à un poste réel, il devient beaucoup plus familier avec son environnement de travail. Le risque de faire une manœuvre erronée sur le mauvais équipement et d'entraîner des conséquences désastreuses est alors diminué. Les erreurs de manœuvres peuvent être

coûteuses pour une entreprise de transport et de distribution d'électricité comme Hydro-Québec.

Avec une formation assistée par un simulateur de postes et de manœuvres en réalité virtuelle, l'opérateur est également moins intimidé par les appareils du poste. En effet, sa formation permet de travailler sur une représentation de ces mêmes appareils. Conséquemment, l'opérateur réagit plus efficacement et plus rapidement lorsqu'il a à faire face à des situations d'urgence car il est déjà familier avec les appareils du poste. La formation devient aussi beaucoup plus flexible puisqu'elle s'adapte facilement aux changements apportés à un poste. Par exemple, lorsqu'un nouveau poste est bâti ou qu'une nouvelle technologie est ajoutée à un poste existant, les opérateurs peuvent s'entraîner dans une reproduction de ce poste avant même la fin des travaux de construction ou de modernisation. Le temps de rodage d'un nouveau poste, où les opérateurs se familiarisent avec l'équipement, est alors pratiquement éliminé.

Contribution personnelle à cette recherche

L'ampleur de cette recherche est trop grande pour être réalisée par une seule personne. Un total de six personnes ont été impliquées directement dans ce projet: trois de l'Université McGill et trois de l'École Polytechnique de Montréal. Les étudiants de l'Université McGill étaient responsables de l'interaction avec le poste virtuel, de la commande vocale, du support multimédia ainsi que de la réalisation de la salle de

commande du poste en 3D. L'intégration du logiciel OPÉRATEUR+, la conversion des schémas unifilaires du logiciel OPÉRATEUR+ en une représentation 3D de même que l'architecture initiale du simulateur de postes ont été la responsabilité des étudiants de l'École Polytechnique. Étant moi-même responsable de l'intégration d'OPÉRATEUR+ au simulateur, l'objectif personnel de ce mémoire est *d'intégrer au simulateur de postes en réalité virtuelle, ESOPE-RV, les deux engins du valideur de manœuvres d'OPÉRATEUR+, soit l'engin de calcul et l'engin de validation de manœuvres*. Cette intégration a nécessité le développement des éléments suivants :

- La création d'un processus nommé ESOPE, qui joue un rôle d'interprète entre le valideur de manœuvres et ESOPE-RV;
- La mise au point d'un langage de communication entre ESOPE et ESOPE-RV qui donne à ce dernier un accès facile aux deux engins du valideur de manœuvres, malgré les systèmes d'exploitation différents utilisés par le valideur de manœuvres et par ESOPE-RV;
- La mise au point d'une méthode de gestion des accès aux deux engins du valideur de manœuvres;

Ce mémoire présente donc les différents choix effectués lors de l'intégration de ce logiciel. De plus, l'architecture du simulateur ESOPE-RV est également présentée dans ce mémoire. La conception de cette architecture est de Jean-Claude Rizzi et d'Étienne Garant mais j'ai été impliqué activement dans son déploiement.

Conclusion

La problématique que ce travail vise à résoudre a été présentée en guise d'introduction à ce mémoire. Un environnement de travail complexe et souvent dangereux, de même qu'une formation adéquate mais très générale, suggèrent que de nouveaux outils de formation flexibles et réalistes sont maintenant nécessaires. La solution proposée, le simulateur ESOPE-RV, consiste à simuler en réalité virtuelle une installation et de permettre à l'opérateur d'y circuler et de réaliser des plans de manœuvres réalistes. Le prochain chapitre présente les différents éléments du simulateur ESOPE-RV.

2. Description du simulateur ESOPE-RV

2.1 *Introduction*

Le simulateur ESOPE-RV permet essentiellement de réaliser un plan de manœuvres à partir d'une représentation en réalité virtuelle d'un poste. Ce simulateur est cependant composé de plusieurs éléments très intéressants. En effet, il comprend également une interface par reconnaissance vocale, un convertisseur en 3D de schémas unifilaires et un environnement multimédia riche. Il utilise des périphériques d'entrée originaux et il intègre un logiciel commercial de validation de manœuvres, le valideur de manœuvres d'OPÉRATEUR+. Ces éléments sont présentés dans ce chapitre.

2.2 *Représentation 3D de l'installation*

2.2.1 *Choix d'un outil de création d'environnement virtuel*

La représentation en réalité virtuelle d'un environnement est essentiellement un problème d'ordre graphique. Les pixels à l'écran de l'ordinateur doivent être positionnés de façon à représenter fidèlement l'environnement réel et à donner une sensation de profondeur à l'environnement. Pour représenter de façon réaliste un poste électrique, il faut que les lignes, les transformateurs, la salle de commande et les autres éléments physiques du poste soient reproduits à l'écran. Il est également nécessaire

que les appareils qui sont normalement sur le sol le soient réellement et que le rapport des dimensions entre les éléments soit respecté. Finalement, parce que la personne peut se déplacer dans cet environnement virtuel, la perspective du poste visible à l'écran doit changer suite à chaque déplacement. Par exemple, on doit pouvoir aller voir derrière un appareil du poste.

La création des différents appareils ainsi que la gestion de l'environnement virtuel peut s'accomplir de deux façons: en achetant un outil qui permet de le faire pour soit ou en créant les fonctions et les routines qui dessinent à l'écran des polygones, qui les positionnent dans un environnement 3D, qui gèrent les déplacements de l'utilisateur dans l'environnement, etc. Cette seconde solution demande près d'un an de travail avant d'avoir les outils permettant de construire en réalité virtuelle un poste électrique. La première solution rend disponible immédiatement les outils nécessaires à la création d'un poste en réalité virtuelle. Le but de ce travail étant de démontrer que la réalité virtuelle est un atout pour la formation des opérateurs, la première solution a été adoptée.

Il existe plusieurs choix d'outils de création d'environnement virtuel qui permettent de créer des objets 3D et de gérer l'interaction entre les objets et l'utilisateur. Les plus populaires sont: WorldToolKit disponible pour PC et SGI (SENSE8, 1997), MR Toolkit pour SGI seulement (MRToolkit, 1997) et REND386 ainsi que les produits dérivés de REND386 (VR386 et Avril) pour PC seulement. Le premier coûte environ

5000\$ US alors que les deux autres sont gratuits. WorldToolKit a été choisi principalement parce qu'il est plus complet que les deux autres et parce qu'il supporte un nombre plus élevé de périphériques. De plus, les ordinateurs prêtés par Hydro-Québec afin de réaliser le projet sont des postes de travail fabriqués par Silicon Graphics Inc (SGI) de Mountain View, CA, USA, et sont supportés par une version de WorldToolKit pour ce type d'ordinateur. Le tableau 2-1 résume les avantages et les inconvénients de ces trois outils de création d'environnement virtuel.

Tableau 2-1 Comparaison des outils

	Avantages	Inconvénients
WorldToolkit SENSE8 Corporation, Mill Valley, CA, USA	<ul style="list-style-type: none"> • disponible pour PC et SGI • supporte plusieurs périphériques dont les lunettes CrystalEyes • souple car il est entièrement programmable • support technique • haute résolution (1280X1024) • supporte les textures 	<ul style="list-style-type: none"> • demande de la programmation
MR Toolkit University of Alberta Edmonton, Al, Canada	<ul style="list-style-type: none"> • gratuit aux écoles • code source disponible • disponible pour SGI et RS6000 	<ul style="list-style-type: none"> • disponible seulement pour SGI • support technique sur internet seulement • peu flexible • pas de support pour lunettes CrystalEyes
REND386 (VR386 et Avril) Sites FTP : psdych.toronto.edu	<ul style="list-style-type: none"> • gratuit • disponible pour PC, une plate-forme peu coûteuse 	<ul style="list-style-type: none"> • support technique sur internet seulement • basse résolution (300X240) • peu flexible • seulement pour DOS • développement interrompu

2.2.2 WorldToolKit de Sense8

WorldToolKit est avant tout une librairie de fonctions pouvant être utilisé dans un programme en C ou en C++. Il existe des fonctions pour créer l'environnement 3D, pour créer des objets 3D et les positionner, pour ajouter ou déplacer une source lumineuse, pour gérer les différents périphériques, etc. Une trajectoire préétablie peut être imposée aux objets mobiles. Il est également possible d'importer des objets 3D créés avec le populaire logiciel 3DSTUDIO de Alias/WaveFront (Toronto, Ont, Canada).

Un aspect intéressant de WorldToolKit est qu'il peut être utilisé avec des lunettes à cristaux liquides. Ces lunettes, lorsque combinées avec un outil graphique pour réalité virtuelle qui peut afficher alternativement deux images avec une perspective légèrement différente (comme WorldToolKit), permettent d'avoir une vue stéréoscopique de l'environnement virtuel. Le simulateur ESOPE-RV utilise les lunettes CrystalEyes de StereoGraphics Corp. (San Rafael, Ca, USA) pour donner un effet de profondeur (3D) au poste virtuel présenté à l'écran.

2.2.3 Création et gestion de l'environnement 3D

L'installation est modélisée sur un écran d'ordinateur de grande dimension (environ 21 pouces) et de façon à ressembler à un poste réel. Malheureusement, ce type de

représentation n'offre pas le plus haut degré d'immersion possible. Cette approche est toutefois peu coûteuse et permet tout de même de démontrer la faisabilité du système de formation d'opérateurs en réalité virtuelle.

Après le démarrage du simulateur 3D, le schéma unifilaire du valideur de manœuvres est interrogé afin d'obtenir le nom des appareils, leurs caractéristiques, etc. La conversion vers une représentation 3D est ensuite effectuée. Une fois le schéma converti, la mise en place de l'environnement 3D débute. Chaque appareil du poste est créé en trois dimensions puis positionnés dans l'environnement 3D. Les figures 2-1 et 2-2 présentent l'extérieur du poste de même que la salle de commande du poste virtuel de formation.

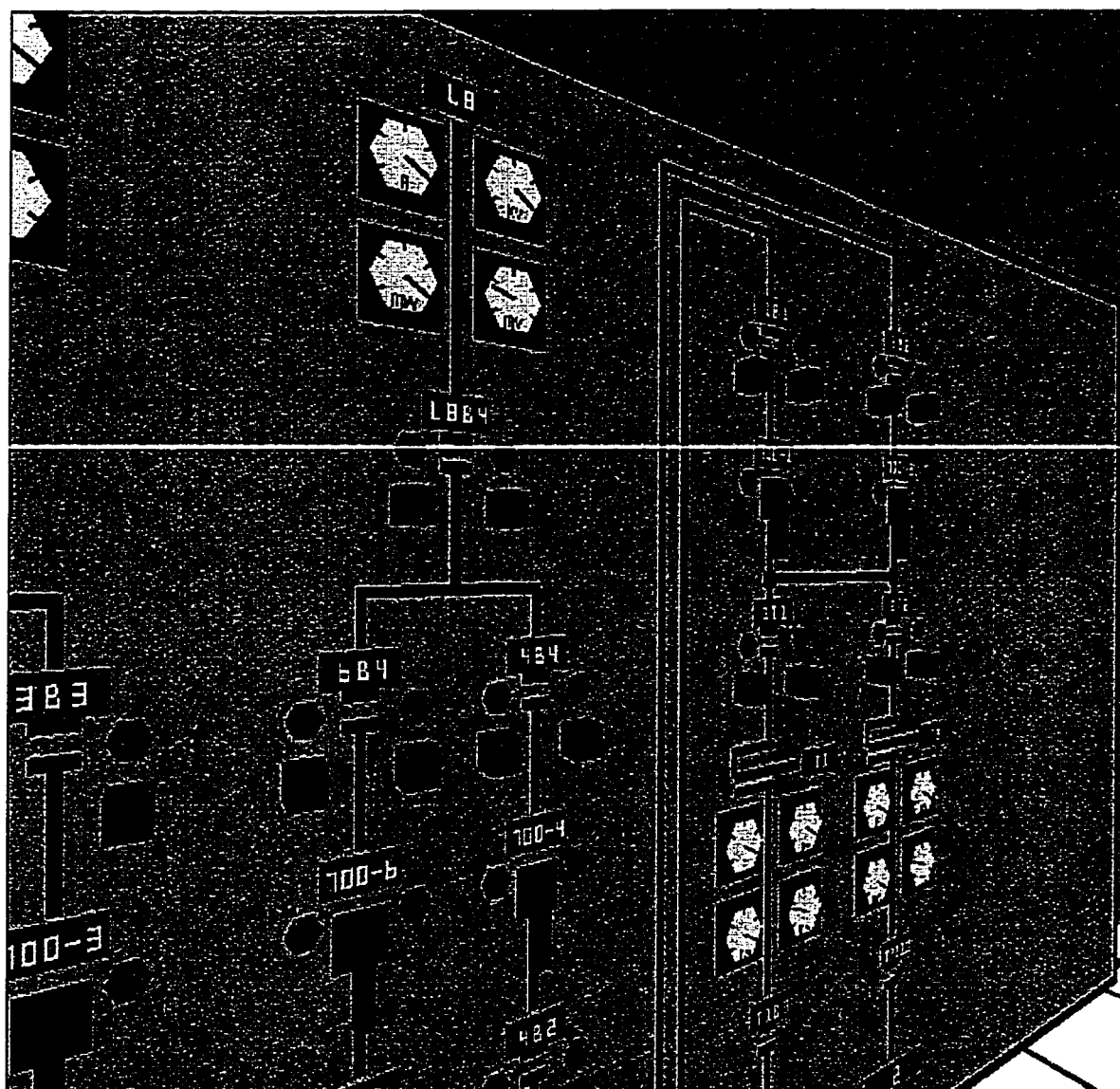


Figure 2-1 Panneau de la salle de commande

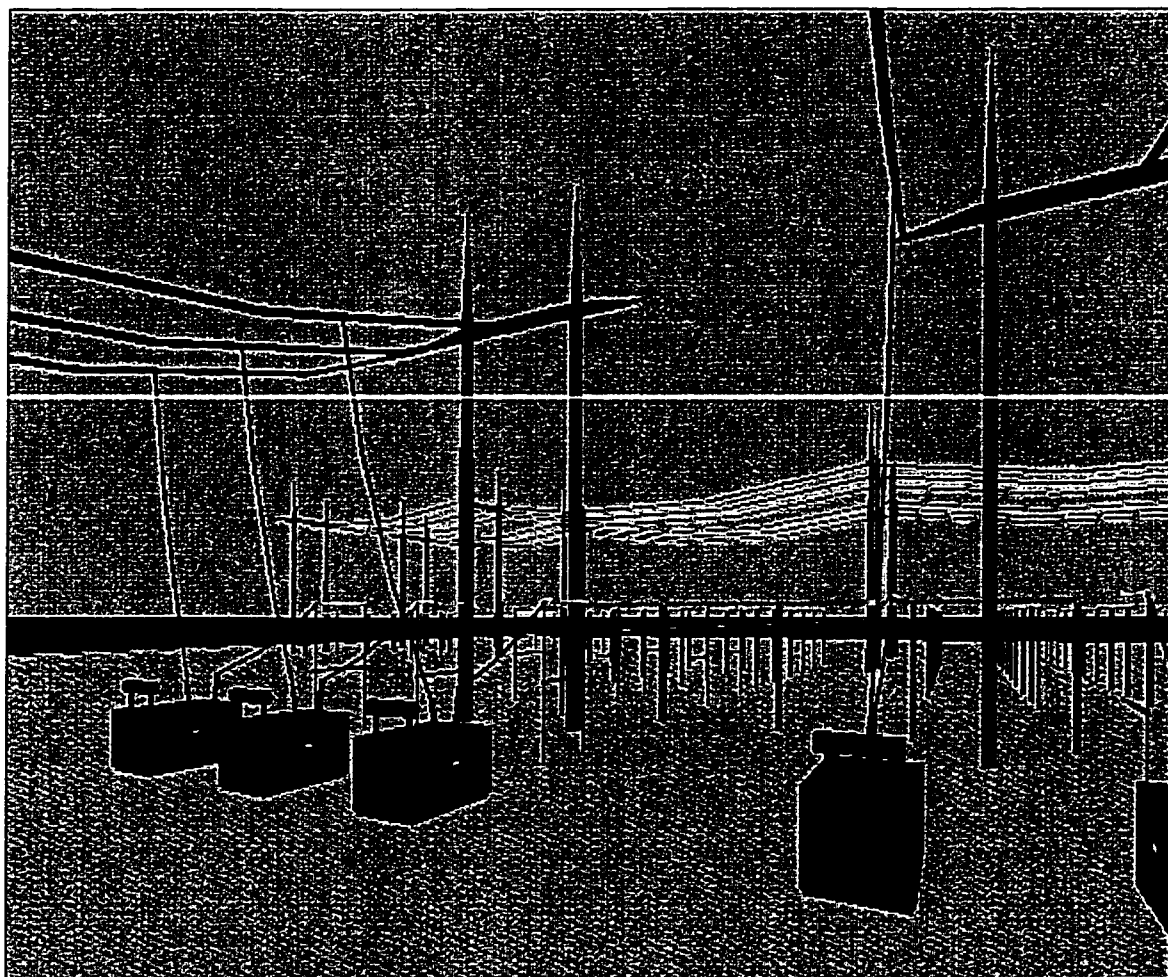


Figure 2-2 Extérieur du poste

L'environnement 3D étant créé, la simulation peut commencer. L'opérateur en formation débute habituellement la simulation à l'extérieur du poste. Il se déplace en cliquant avec la souris dans le haut ou le bas de l'écran, pour avancer et reculer respectivement dans le poste. L'opérateur peut également tourner vers la gauche ou la droite en cliquant dans la partie gauche ou droite de l'écran. Il peut ainsi se déplacer autour des appareils au sol et sous les appareils rattachés aux structures métalliques du poste. Les lunettes à cristaux liquides CrystalEyes permettent de donner un effet de profondeur aux éléments de l'installation virtuelle.

Pour aller à l'intérieur de la salle de commande, l'opérateur n'a qu'à se déplacer près de l'entrée du bâtiment et de simplement passer par la porte. Quelques secondes plus tard, l'opérateur est à l'intérieur de la salle de commande. Le passage vers l'extérieur s'effectue également par la porte du bâtiment. La salle de commande contient trois panneaux de commandes: le panneau de changement d'état du côté haute tension du poste, le panneau de changement du mode de commande (du mode local au mode télécommande ou l'inverse) du côté haute tension ainsi que le panneau basse tension du poste. Dans un poste réel, les éléments des deux premiers panneaux sont habituellement sur un seul panneau. Ils ont été séparés en deux afin de diminuer la quantité d'éléments affichés simultanément car la vitesse de la simulation se trouve à augmenter lorsqu'on diminue la quantité d'éléments affichés. Chaque panneau contient des bouton-poussoirs, des leviers et des témoins lumineux (rouge et vert). Les témoins et les leviers représentent l'état réel du paramètre de l'appareil.

L'opérateur peut réaliser une manœuvre en actionnant les leviers et les boutons-poussoirs des panneaux du côté haute tension (figures 2-3 et 2-4).

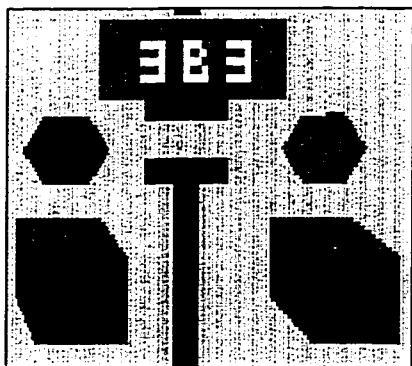


Figure 2-3 Boutons du panneau de commande

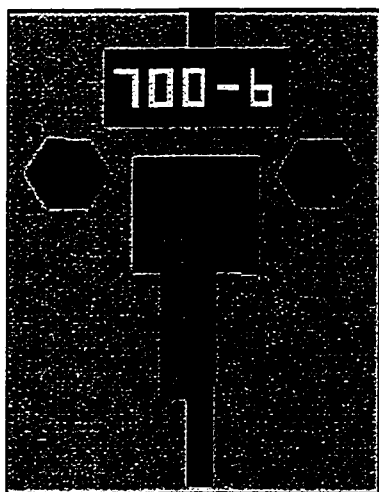


Figure 2-4 Manette du panneau de commande

Bien qu'il soit possible d'actionner un bouton-poussoir ou un levier directement par une commande vocale, la méthode présentée dans cette section utilise la main virtuelle pour actionner ces éléments (figure 2-5). La main est affichée à l'écran à l'aide d'une commande vocale (voir la section 2.5 pour plus d'informations sur la commande vocale).

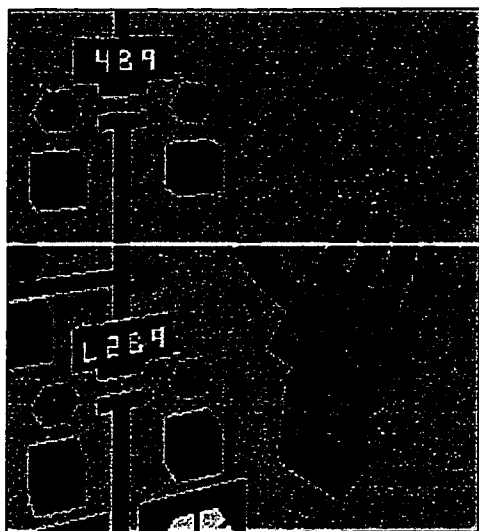


Figure 2-5 Main virtuelle devant le panneau de commande

Il faut ensuite toucher à l'élément qu'on désire actionner avec la main virtuelle. La main peut être déplacée en utilisant la souris, de la même façon que l'opérateur se déplace. Le gant peut également être employé pour déplacer la main virtuelle. Le fait de toucher à un bouton-poussoir ou à un levier au moyen de la main virtuelle soumet la manœuvre associée au valideur de manœuvres afin d'être validée. Par exemple, en actionnant la manette représentant l'état d'un disjoncteur, la manœuvre consistant à

changer l'état de ce disjoncteur est soumise au valideur de manœuvres pour acceptation.

Si la manœuvre est acceptée, le levier change de position et les témoins lumineux changent de couleur afin d'indiquer le nouvel état de l'appareil. Le bouton-poussoir donne l'impression d'être pressé puis relâché, tout en laissant entendre par les haut-parleurs le clic habituellement associé à ce type de bouton. Les manœuvres interdites laissent entendre un message audio explicatif de l'erreur commise par l'opérateur.

Divers mécanismes ont été ajoutés afin d'accélérer le rythme de la simulation en réalité virtuelle. Le plus intéressant est l'utilisation de différents niveaux de détails dans la représentation des appareils. Lorsque l'opérateur est loin d'un appareil, ce dernier n'est pas affiché ou il est représenté que par un cube. À mesure que l'opérateur s'approche de l'appareil, l'objet 3D représentant l'appareil devient de plus en plus détaillé. Cette technique, employée fréquemment dans les jeux vidéos, permet de diminuer le nombre de polygones affichés simultanément à l'écran et d'accélérer le rythme de la simulation.

2.3 Logiciel OPÉRATEUR+

La conception et le développement du valideur de manœuvres d'OPÉRATEUR+ a débuté en 1987 chez Hydro-Québec. Ce logiciel emploie Microsoft Windows (3.1 ou

95) et possède une interface composée de menus et de fenêtres ainsi qu'une barre d'outils. Il permet de valider les manœuvres effectuées sur les appareils d'un poste. Le poste ou la centrale est représenté à l'écran par son schéma unifilaire (voir la figure 2-6)

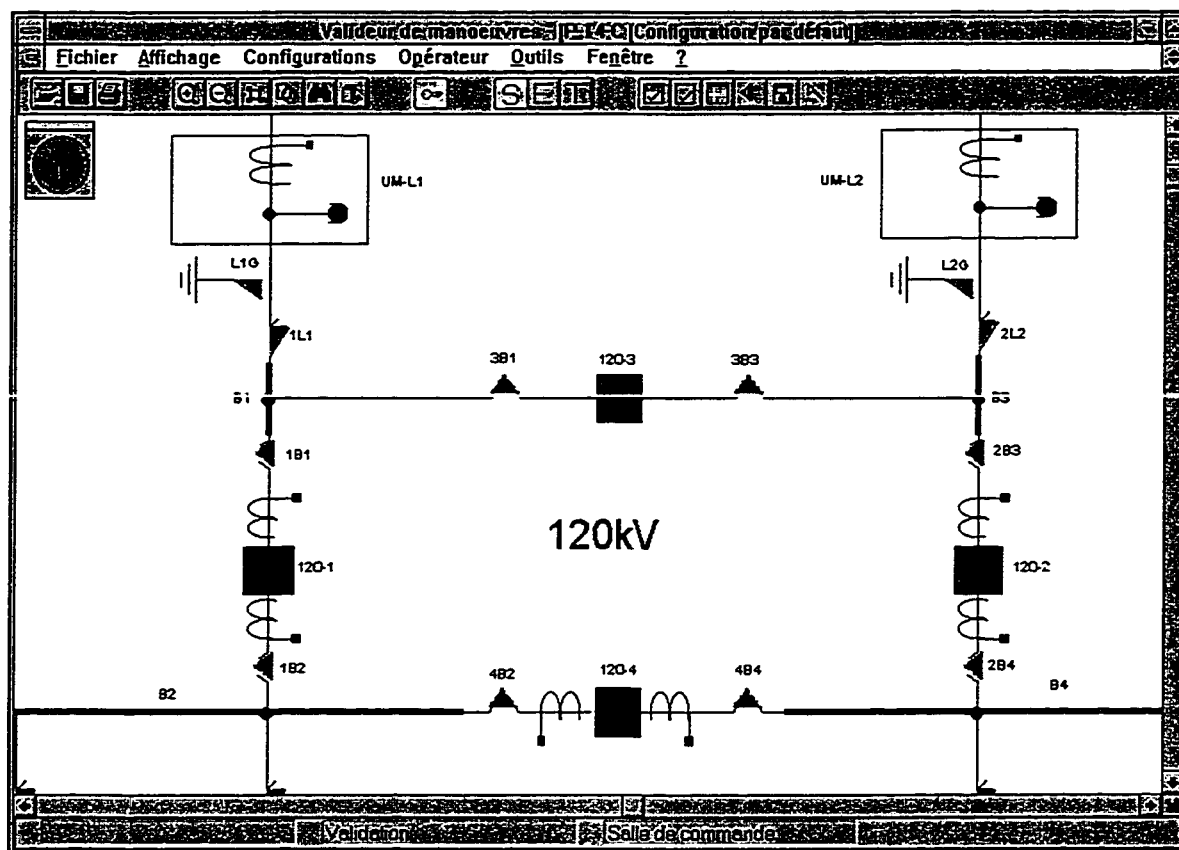


Figure 2-6 Le valideur de manœuvres

Le valideur de manœuvres assiste l'utilisateur lors de la réalisation d'un plan de manœuvres. Chaque action effectuée par l'utilisateur est validée par le logiciel. La validation se fait par l'engin de validation de manœuvres en consultant les règles d'exploitation, emmagasinées dans un fichier. Les règles d'exploitation définissent

l'ensemble des manœuvres qu'un opérateur peut effectuer ainsi que la manière dont celles-ci sont réalisées. Le valideur de manœuvres contient également un engin de calculs qui permet de connaître les tensions et les courants sur les appareils du schéma unifilaire, en prenant en considération l'impédance des appareils.

Un aspect très intéressant du valideur de manœuvres est que ce dernier a déjà un lien avec l'extérieur: il peut être branché à un SCADA qui lui permet d'acquérir les paramètres exacts des appareils d'une installation correspondant à un schéma unifilaire. Cependant, la communication ne se fait que dans un sens: si l'utilisateur change l'état d'un appareil dans le schéma unifilaire du valideur de manœuvres, le même appareil dans l'installation n'est pas modifié.

2.4 Conversion de schémas unipolaires

La conversion d'un schéma unifilaire vers sa représentation 3D est une tâche complexe. Le schéma unifilaire contient très peu d'information sur les caractéristiques physiques des appareils. Il donne un indice quant à la disposition générale des appareils mais il ne dit rien sur la dimension des appareils, la hauteur des appareils, la présence de structures métalliques agissant comme support, etc. De plus, le schéma ne représente qu'une phase électrique alors qu'en réalité, les trois phases sont présentes dans un poste.

La base de la stratégie de conversion est la configuration des barres du poste (Garant, Daigle et al., Juin 1995). Il existe six (6) configurations principales de barres: *double bus single* ou *double breaker*, *Breaker and a half*, *Ring bus*, *Main & transfer bus*, *Single bus* et *Breaker and a half* (Dorf, 1993). Dans un premier temps, l'information sur tous les appareils du poste est obtenue. Ensuite, on trouve la configuration de barres se rapprochant le plus possible de la configuration du schéma unifilaire. Finalement, on calcule la position dans l'espace des différents équipements, en respectant certains critères utilisés lors de la construction de postes, notamment les distances entre les appareils pour chaque niveau de tension. Le point d'ancrage (au sol ou sur une structure) des appareils est également déterminé. Les objets 3D représentant les appareils peuvent maintenant être créés puis positionnés dans l'environnement virtuel, à l'endroit calculé.

Pour plus de flexibilité, le processus de conversion a été fait de façon à être indépendant de l'outil de création d'environnement virtuel utilisé pour construire le poste en 3D (Garant, Daigle et al., Juin 1995). Dans un premier temps, on détermine la position relative de chaque appareil ou élément du poste. Puis, les positions sont transposées vers les positions prescrites par WorldToolKit.

2.5 Commande vocale

La commande vocale est le principal moyen d'interaction entre l'opérateur et le simulateur ESOPE-RV. Le principal avantage de ce moyen d'interaction est de libérer les mains de l'opérateur (Okapuu-von Veh, 1996). Il peut alors utiliser ses mains pour naviguer, pour manipuler la main virtuelle à l'intérieur de la salle de commande, etc.

Le système de reconnaissance vocal employé est le Verbex 7000 (Verbex Voice System, 1990). Il est contenu dans un boîtier de la dimension d'un ordinateur de table. Des connecteurs permettent de brancher un micro et des écouteurs au système de reconnaissance vocal. Ce dernier possédant également un port de communication série, la communication entre le système de reconnaissance vocale et les autres processus du simulateur ESOPE-RV se fait par ce port. Le système Verbex est de type dépendant de l'utilisateur : chaque utilisateur doit entraîner le système afin que ce dernier puisse reconnaître son timbre de voix.

La commande vocale permet de réaliser toutes les manœuvres possibles sur les appareils de l'installation virtuelle, tout comme la main virtuelle. Cependant, l'usage de la commande vocale ne se limite pas aux manœuvres. En effet, il est possible de se déplacer, de se téléporter d'un endroit à l'autre, de vérifier certains aspects d'un appareil ou de l'installation et d'accéder aux photos et aux vidéos de l'aide multimédia.

L'utilisation de la commande vocale est simple: des phrases relativement complètes sont employées. Par exemple, pour ouvrir le disjoncteur 700-3, il suffit de dire « Ouvrir disjoncteur 700-3 » alors que la commande vocale « Téléporter salle de commande » déplace immédiatement l'opérateur de l'extérieur vers la salle de commande. Si l'opérateur ne se souvient pas d'une commande vocale, il peut toujours accéder à une aide en ligne. Il suffit de donner la commande « Aide » pour entendre un message enregistré énonçant les différentes commandes vocales possibles.

2.6 Environnement sonore

Bien que la représentation visuelle à elle seule procure une sensation d'immersion, l'ajout de sons augmente de beaucoup le réalisme de la simulation. Le simulateur ESOPE-RV emploie deux paires de canaux audio pour générer les différents sons (une paire est constituée d'un canal droit et d'un canal gauche). Une paire est utilisée pour reproduire le bruit des transformateurs dans le poste alors que les bruits résultant d'une action de l'opérateur (ouverture d'un disjoncteur, pression sur un bouton-poussoir, etc.) sont générés sur la deuxième paire de canaux audio (Okapuu-von Veh, 1996). L'intensité du bruit des transformateurs, le « hum » (bruit sourd produit par les transformateurs de puissance), dépend de la distance qui les séparent de l'opérateur.

2.7 Support multimédia

L'opérateur dispose d'une banque d'images et de séquences vidéo qu'il peut consulter. Les photos représentent des appareils semblables à ceux présents dans le poste virtuel alors que les séquences vidéo montrent ces mêmes appareils lors de la réalisation de vrais manœuvres. Cependant, ces photos et ces vidéos ne sont pas affichés sur le même écran que la simulation en réalité virtuelle. Parce que des lunettes à cristaux CrystalEyes sont utilisées, les images et les vidéos d'appareils à l'écran sont embrouillées. Un second écran, situé à coté de l'opérateur, permet à ce dernier de visualiser correctement les photos et les séquences vidéo (figure 2-7)

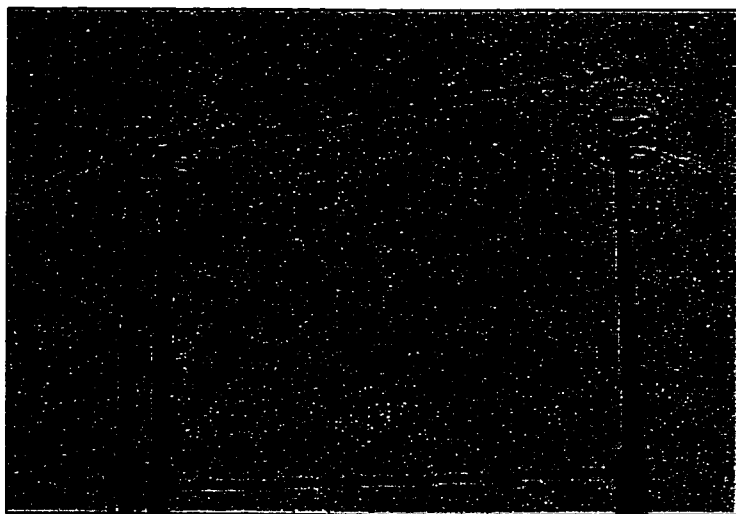


Figure 2-7 Extrait d'une séquence vidéo sur l'ouverture d'un sectionneur à perche

Afin d'accéder à ce support multimédia, la commande vocale est également utilisée. Par exemple, pour visionner le sectionneur 3B3 en action, il suffit de donner la

commande « Vidéo sectionneur 3B3 ». La séquence vidéo apparaît quelques secondes plus tard à l'écran. L'opérateur peut ensuite contrôler la visualisation du vidéo (avance rapide, reculer, pause, jouer) à sa guise également par des commandes vocales.

2.8 Périphériques d'entrée

L'opérateur qui utilise le simulateur en réalité virtuelle dispose de deux périphériques: la souris 2D fournie avec les postes de travail SGI et le gant PowerGlove de Mattel. La souris à deux ou trois boutons est le périphérique privilégié dans le simulateur. La souris permet, en déplaçant le pointeur à l'écran puis en cliquant, de se déplacer et de manipuler la main virtuelle afin de réaliser des manœuvres. L'écran est divisé en quatre régions, chacune d'elles étant associées à un mouvement particulier (figure 2-8)

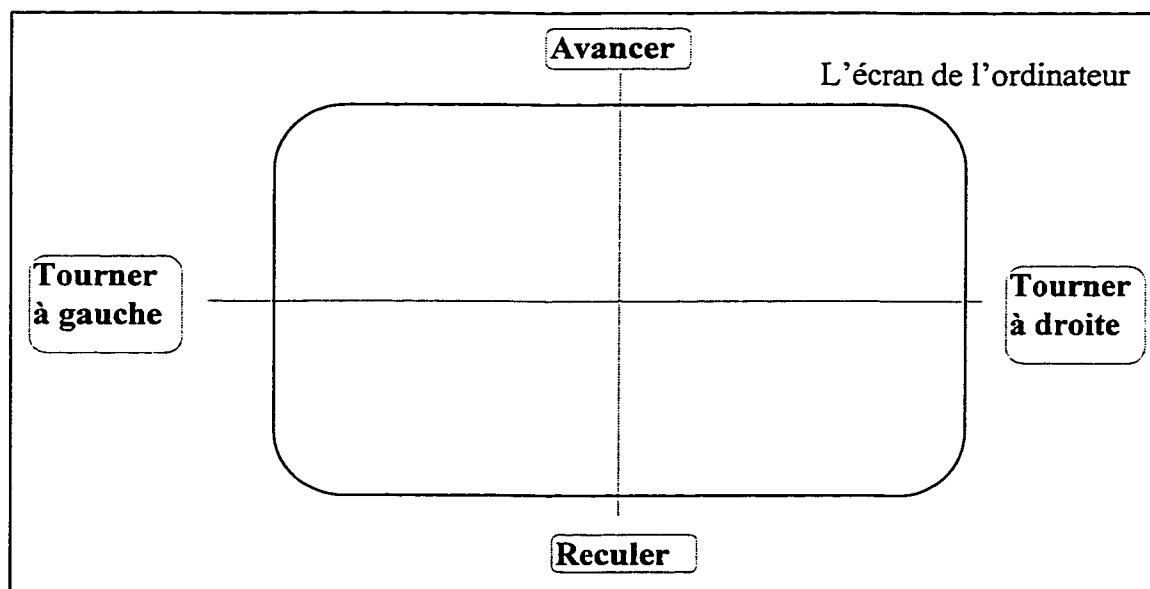


Figure 2-8 Déplacement avec la souris 2D

Par exemple, pour avancer rapidement dans le poste virtuel, il suffit de pointer dans la partie supérieur centrale de l'écran, d'appuyer sur le bouton droit de la souris et de maintenir ce bouton enfoncée.

Le gant permet également de se déplacer et de réaliser des manœuvres. Bien que la manipulation de la main soit facile avec le gant, la navigation dans le poste pose certains problèmes. Par exemple, l'opérateur avance le gant vers l'avant pour avancer. Après quelques secondes, l'opérateur est obligé de se lever de sa chaise et de marcher vers l'avant. Il risque ensuite de se buter à un mur. Afin de lui éviter de se déplacer, l'opérateur peut désactiver le gant, repositionner le gant près de lui, réactiver le gant et déplacer le gant de nouveau vers l'avant afin de poursuivre son déplacement dans cette direction. Pour cette raison, le temps nécessaire pour se déplacer avec le gant est beaucoup plus long que si la souris était utilisée (Shaikh, Garant, Okapuu-von veh, Malowany, Daigle, Desbiens, Rizzi, Marceau, Gauthier, 1995).

La combinaison idéale semble l'utilisation de la souris 2D pour se déplacer et du gant pour manipuler la main virtuelle (Shaikh, Garant, Okapuu-von veh, Malowany, Daigle, Desbiens, Rizzi, Marceau, Gauthier, 1995).

2.9 Configuration et démarrage

Le projet ESOPE-RV est composé de plusieurs processus s'exécutant sur différents ordinateurs. Un fichier de configuration permet de modifier facilement l'endroit d'exécution des différents processus du simulateur de poste. Ce fichier, qui comprend deux parties distinctes, est un simple fichier texte et peut donc être édité à l'aide d'un simple éditeur de texte (ex: emacs, vi, etc.). La première partie de ce fichier contient le nom des processus et l'endroit où s'exécute chacun de ces processus. Le nom d'un processus est habituellement le nom du fichier qui contient l'exécutable. L'endroit d'exécution est définie par le nom du poste de travail. La figure 2-9 contient la première partie d'un fichier de configuration.

<u>ID</u>	<u>MACHINE</u>	<u>PATH</u>	<u>NOM PROCESSUS</u>	<u>PAUSE</u>	<u>EXEC</u>	<u>PARAM</u>
1	ohm	alpha	win-esope	0	no	
2	faraday	alpha	bgsnd	5	yes	
3	fortescue	alpha	showmov	5	yes	
4	fortescue	alpha	audiovid	5	yes	
5	fortescue	alpha	esope	10	yes	
6	mcgillis	alpha	esope-rv	10	yes	
7	fortescue	alpha	verbex	5	yes	
8	mcgillis	alpha	worldtoolkit	5	term	

Figure 2-9 Première partie du fichier de configuration

Par exemple, le fichier indique que le processus **esope-rv**, qui a le numéro 6 comme identificateur, est exécuté sur l'ordinateur portant le nom McGillis. Le processus se

trouve dans le sous répertoire /alpha. Après le démarrage, une pause de 10 secondes est nécessaire pour que le processus puisse correctement effectuer son initiation.

La seconde partie du fichier de configuration décrit les relations entre chacun des processus. Entre deux processus, la relation est la suivante: un processus joue le rôle du client alors qu'un autre est le serveur. Lorsqu'un processus possède des liens de communications avec plus d'un processus, il peut exercer le rôle d'un client et celui d'un serveur. Dans cette situation, il se considère le client d'un processus et le serveur d'un troisième processus. Le rôle d'un processus détermine qui sera l'instigateur du lien de communication. Mais lorsque le lien est établi, le traitement est alors le même pour chacun des processus reliés. La figure 2-10 contient la deuxième partie du fichier de configuration.

4	client	2	serveur
4	client	3	serveur
5	client	4	serveur
6	client	4	serveur
6	client	5	serveur
7	client	4	serveur
7	client	5	serveur
7	client	6	serveur
5	client	1	serveur

Figure 2-10 Seconde partie du fichier de configuration

Dans ce tableau, on remarque que pour le lien qui relie les processus 2 et 4 (**bgsnd** et **audiovid** respectivement), le processus 4 est le client alors le processus 2 est le serveur.

Les différents rôles que peuvent prendre chacun des processus impliquent une séquence de démarrage fixe car le serveur est toujours l'instigateur d'un lien. L'objectif de cette séquence est de s'assurer que le serveur s'exécute avant chacun de ses clients. Lorsque l'architecture est composée de seulement deux processus, cette séquence ainsi que sa synchronisation est assez triviale: le serveur est lancé suivi peu de temps après par le client. Cette même séquence est plus difficile à déterminer pour le simulateur ESOPE-RV car le nombre de processus est élevé. La séquence de démarrage est spécifiée dans la première partie du fichier de configuration. La séquence est l'ordre dans lequel les processus sont énumérés. Selon la figure 2-9, la séquence de démarrage des processus débute par le processus **win-esope** (le valideur de manœuvres d'OPÉRATEUR+). Cependant, ce processus n'a pas à être démarré car le champ EXEC est à NO. En effet, avant d'exécuter le script de démarrage, le valideur de manœuvres doit être lancé manuellement car ce dernier ne peut être démarré à distance. Le processus suivant, **bgsnd**, est donc le premier à être démarré. Après un délai de cinq secondes, le prochaine processus, **showmov**, est démarré. Les autres processus sont tous démarrés de cette manière, jusqu'au dernier, le processus **worldtoolkit**.

Le fichier de configuration est lu par chaque processus au moment d'établir les différents liens de communication. En effet, la seconde partie de ce fichier indique au processus avec quel processus établir chaque lien ainsi que son rôle (client ou serveur). La première partie n'est cependant lue par aucun processus: cette partie n'est utile qu'au script de démarrage. Ce script, écrit dans le langage PERL, lance l'exécution de chacun des processus sur son poste de travail, dans la séquence indiquée dans le fichier de configuration. Ce script à l'avantage de pouvoir être lancé de chacune des stations de travail. Le script utilise les outils d'UNIX permettant de lancer des applications sur d'autres stations de travail. Pour permettre au système d'exploitation de charger le processus en mémoire et de l'exécuter, un délai de quelques secondes sépare le lancement de chaque processus. Lorsque l'exécution se fait sur une autre station, un délai plus élevé peut être nécessaire afin de compenser pour les accès réseaux plus lent. La syntaxe permettant l'exécution du script et le démarrage d'ESOPÉ-RV est: `erv start`. Il est aussi possible de mettre fin à l'exécution du simulateur en appelant le même script de la façon suivante: `erv stop`. Le script arrête alors l'exécution de chacun des processus du simulateur.

Pendant l'exécution du script, divers messages affichés à l'écran permettent à l'utilisateur de suivre le démarrage du simulateur de poste. L'information disponible comprend le nom des processus, l'initiation des différents liens et le nom des stations de travail. La figure 2-11 présente une partie des messages affichés à l'écran.

```

1.  *** SKIP: win-esope ***
2.  rsh faraday -n /home/samedi2/ami/esoperv/demopoly/src/alpha/bgsnd &
3.  BCK >>>> A_V
4.  Bgsound serveur process on faraday
5.  ... waiting for distant connection from AudioVideo on fortscue
6.  /home/samedi2/ami/esoperv/demopoly/src/alpha/showmov &
7.  MOV >>>> A_V
8.  Showmovie serveur process on fortscue
9.  ... waiting for local connection from AudioVideo on fortscue
10. rsh fortscue -n /home/samedi2/ami/esoperv/demopoly/src/alpha/audiovid &
11. A_V >>>> BCK
12. BCK <<<< A_V
13. MOV <<<< A_V
14. AudioVideo client process on fortscue
15. ... waiting for distant connection from Bgsound on faraday
16. A_V <<<< BCK
17. A_V >>>> MOV
18. AudioVideo client process on fortscue
19. ... waiting for local connection from Showmovie on fortscue
20. A_V <<<< MOV

```

Figure 2-11 Messages affichés lors du démarrage

Le message 1 indique que le processus **win-esope** (le valideur de manœuvres) n'est pas démarré par le script. Les messages 2, 6 et 10 indiquent que les processus **bgsnd**, **showmov** et **audiovid** ont été lancés. Les messages aux lignes 4 et 5 annoncent que le processus a été lancé avec succès et qu'il souhaite établir un lien avec un autre processus. Les messages 8, 9, 14, 15, 18 et 19 sont également du même type. Quant aux messages avec les caractères « >>>> », ils permettent de suivre l'établissement d'une connexion entre deux processus. Par exemple, pour le lien entre **bgsnd** et **audiovid**, il y a quatre messages d'impliqués:

ligne 3: BCK >>>> A_V, bgsnd (BCK) désire établir un lien avec AudioVideo (A_V),

ligne 11: A_V >>>> BCK, AudioVideo(A_V) désire établir un lien avec bgsnd(BCK),

ligne 12: BCK <<<< A_V, le lien entre bgsnd et AudioVideo est établi, selon bgsnd,

ligne 16: A_V <<<< BCK, le lien entre bgsnd et AudioVideo est établi, selon AudioVideo.

Les différents messages ne sont pas toujours dans un ordre synchrone.

2.10 Conclusion

Le simulateur de manœuvres en réalité virtuelle, ESOPE-RV, incorpore plusieurs technologies et techniques afin d'en faire un système de formation complet et réaliste. Il intègre un logiciel de validation de manœuvres existant, l'aide multimédia ainsi qu'une interface de commande vocale. De plus, il utilise un outil de création d'environnement virtuel commercial et un périphérique prometteur pour ce genre d'application, le gant. L'architecture du simulateur de même que les moyens de communication entre les processus n'ont cependant pas encore été précisés. Le prochain chapitre présente l'architecture du système et les moyens de communication.

3. Architecture d'ESOPE-RV

3.1 *Introduction*

L'architecture du simulateur ESOPE-RV est le résultat d'un certain nombre de contraintes, surtout des contraintes matérielles. Malgré le coût élevé d'acquisition des ordinateurs utilisés pour ce projet (entre 20000\$ et 50000\$ selon le modèle d'ordinateur, de 1989 à 1992), la puissance de ces ordinateurs n'est pas suffisante pour les besoins du projet. L'architecture présentée dans ce chapitre tente de maximiser les performances du simulateur en utilisant de façon efficace et judicieuse la puissance disponible des ordinateurs. Les éléments physiques, de même que les moyens de communication employés, sont également présentés.

3.2 *Description des éléments physiques*

Le matériel informatique utilisé pour la réalisation de ce projet est composé de stations de travail Indigo de Silicon Graphics (SGI). Deux ordinateurs de type PC sont aussi utilisés pour l'exécution de la reconnaissance vocale et du valideur de manœuvres, OPÉRATEUR+. Ce dernier nécessite comme système d'exploitation DOS (version 6.2) et Windows for Workgroups (version 3.11). La reconnaissance vocale nécessite quant à elle un PC avec le système d'exploitation LINUX, qui est une variante disponible gratuitement du populaire système d'exploitation UNIX. En ce qui

concerne les ordinateurs de marque SGI, le système d'exploitation utilisé est IRIX (version 5.2), qui est également une variante du système d'exploitation UNIX. Ce système d'exploitation comprend la plupart des fonctions habituellement disponibles sous UNIX. Par exemple, IRIX supporte l'envoi de messages par sockets, le protocole de communication TCP/IP (Comer, 1991), l'importation et l'exportation de disques (NFS) ainsi que l'exécution simultanée de plusieurs programmes (communément appelée multitâche). Lors de l'exécution du simulateur de postes ESOPE-RV, trois postes de travail sont utilisées, chacune ayant un minimum de 32 mégaoctets de mémoire vive. Plus spécifiquement, un poste de travail est un Indigo R4000 (modèle ELAN) alors que les deux autres sont de types Indigo R3000. Ces postes utilisent des processeurs R3000 et R4000 développés par MIPS en 1988 et 1992 respectivement (Burd, 1997). La puissance de ces stations de travail R3000 se compare à celle des ordinateurs de type PC dotés d'un processeur Intel 80486 (Burd, 1997). La station R4000 est plus puissante et elle est l'équivalent d'un PC doté d'un processeur Pentium 100 Mhz ((Burd, 1997). Le système vidéo de ces postes offre la capacité d'afficher dans une résolution de 1280x1024 avec 256 couleurs sur l'indigo R4000 et une résolution de 1024x768 avec 256 couleurs également pour le R3000. Ces postes n'ont pas de processeurs spécialisés pour l'application de textures sur un polygone ou pour animer un objet 3D à l'écran.

Des tests préliminaires à l'aide d'un logiciel de création d'environnement 3D ont permis de constater la faible puissance de calcul de ces postes de travail pour ce genre

d'application. Le choix de l'architecture du système doit permettre de tirer le maximum de l'équipement informatique disponible. Dans le but d'avoir un simulateur de postes de transport d'électricité en réalité virtuelle réaliste, la condition suivante a été posée: le niveau de complexité de la représentation du poste doit être minimal. De plus, la puissance de calcul du poste de travail le plus puissant doit être réservée à l'exécution de la tâche la plus difficile, soit la représentation en réalité virtuelle d'un poste. La première condition est discutée dans la section 6.2. Le choix de l'architecture du simulateur, présenté dans les prochaines sections, est en partie imposé par la seconde condition.

3.3 Modules et processus de l'architecture

Pour dédier un poste de travail à une tâche particulière, il est nécessaire de regrouper sur celui-ci le traitement requis pour effectuer ce travail. Le traitement non relié à ce travail est quant à lui effectué sur un autre poste. La nécessité d'avoir une architecture comprenant plusieurs processus ou programmes est à présent évidente. Le traitement de la représentation du poste de transport en réalité virtuelle est donc effectué à l'intérieur d'un processus sur le poste de travail le plus performant. Le traitement non relié directement à la représentation 3D du poste est effectué par un autre processus, sur un poste de travail différent. Ces processus ont à leur disposition un moyen de communication permettant d'échanger de l'information entre eux. Cet aspect est discuté dans la prochaine section.

Bien que possible, l'utilisation d'un seul programme pour effectuer l'ensemble des autres tâches engendre de nombreux désavantages. Le niveau de complexité de la programmation est dans ce cas augmenté car plusieurs tâches différentes doivent être gérées simultanément. Certaines tâches plus longues à effectuer ont comme effet de ralentir ou de retarder le traitement des autres tâches. Le programme résultant est aussi plus gros donc plus gourmand en mémoire. L'utilisation de plusieurs processus offre la possibilité de réaliser des programmes plus petits dont la gestion est habituellement plus simple. Pour ces raisons, chaque processus du simulateur effectue le traitement pour une tâche bien spécifique. Cette approche permet aussi de faciliter le développement d'ESOPE-RV car chaque personne impliquée dans le projet peut avoir à sa charge la réalisation d'un programme différent. Puisque chaque personne travaille sur des fichiers sources distincts, la gestion de ces fichiers est grandement simplifiée.

Pour faciliter la compréhension de l'architecture du simulateur, les différentes tâches devant être réalisées pour représenter un poste de transport d'électricité en réalité virtuelle sont regroupées en modules (figure 3-1).

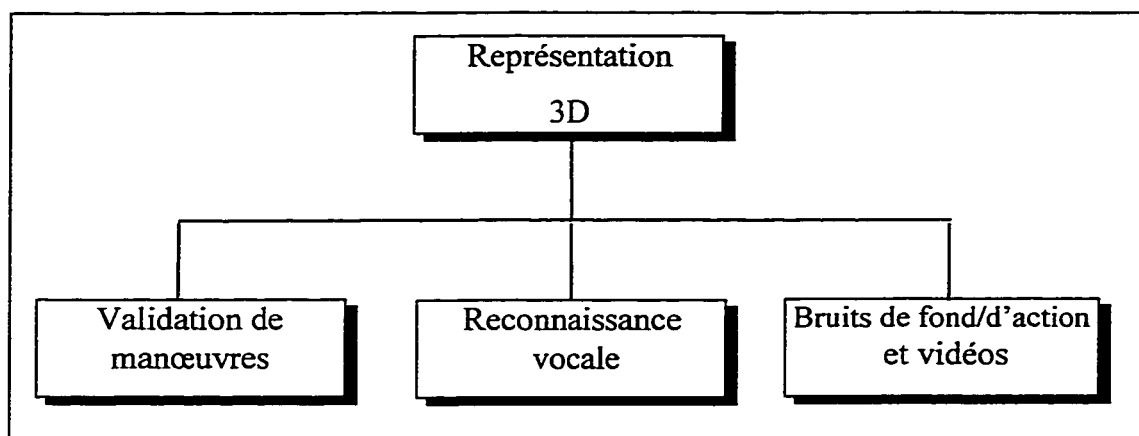


Figure 3-1 Modules dans le simulateur ESOPE-RV

Un module est constitué d'un ou de plusieurs processus qui effectuent un traitement spécifique. Les traits de la figure 3-1 représentent un système de communication permettant aux modules d'échanger de l'information.

Le module de représentation 3D est composé d'un premier processus qui effectue le traitement nécessaire pour représenter en réalité virtuelle le poste sur lequel se déroule la simulation. Ce processus, appelé **worldtoolkit**, a aussi à sa charge le traitement des mouvements de la souris et du gant par l'opérateur en formation. Un deuxième processus, le processus **esope-rv**, est utilisé pour gérer les communications avec les autres modules. Ce second processus permet d'accélérer le déroulement de la simulation 3D. La prochaine section explique en détails le fondement de ce choix.

La validation de manœuvres est aussi un des modules du simulateur. Il est formé de deux processus: le premier, le valideur de manœuvres **OPÉRATEUR+**, contient l'engin de calcul ainsi que l'engin de validation de règles d'exploitation utilisés

pendant une simulation d'un poste. Ce logiciel existait avant la réalisation du projet ESOPE-RV et seulement son intégration au simulateur ESOPE-RV fait partie de ce mémoire. Cette intégration se fait par l'entremise d'un second processus, le processus **esope** qui agit comme interface entre le simulateur ESOPE-RV et le valideur de manœuvres d'OPÉRATEUR+. Cette approche permet d'isoler le simulateur ESOPE-RV des éventuels changements apportés au valideur de manœuvres. En effet, lorsque des modifications sont apportées à ce dernier, elles entraînent une répercussion uniquement au niveau du processus d'interface. Les autres modules sont donc en quelque sorte immunisés contre des modifications dans la façon d'accéder aux deux engins du valideur de manœuvres.

Un module différent s'acquitte de l'aspect multimédia d'une simulation ainsi que de l'environnement sonore. Il est responsable des bruits de fonds, des bruits d'action de même que de l'affichage de photos et de clips vidéo. Les bruits de fond, comme le bruit sourd et monotone produit par un transformateur, sont pour la plupart entendus tout au long d'une simulation. Les bruits d'action sont plus ponctuels : ils reproduisent des bruits d'équipement suite à une manœuvre, comme l'ouverture d'un disjoncteur à air comprimé. Les clips vidéo et les photos représentent de véritables pièces d'équipement sous manœuvre. De plus, ce module lit, à partir du disque rigide, des fichiers contenant des sons, des photos et des vidéos. Ces fichiers sont habituellement volumineux et ils exigent un certain temps avant d'être chargés en mémoire. Le temps requis pour effectuer la lecture de ces fichiers retarderait normalement l'exécution des

autres tâches. Le traitement dans un module séparé dégage donc les autres modules de ce travail ardu et par le fait même, leur permet d'effectuer leur traitement plus rapidement. Trois processus composent ce module. Le premier, **audiovid**, traite les demandes d'audition de bruits et de visualisation de photos/clips vidéo provenant du simulateur ESOPE-RV, effectue une validation de ces demandes et gère les erreurs. Il détermine aussi l'intensité sonore des différents bruits selon la position de l'opérateur dans l'environnement virtuel. Un deuxième processus, appelé **bgsnd**, lit le fichier de format AIFF (Apple, 1991) contenant le bruit de fond et le fait entendre par les haut-parleurs du poste de travail. Un dernier processus, **showmov**, fait la lecture des photos et des clips vidéo.

Le module de reconnaissance vocale est l'interface entre l'appareil effectuant la reconnaissance vocale et le simulateur de postes électriques ESOPE-RV. Cet appareil, de marque Verbex 7000, communique avec un poste de travail par un port de communication série. Un seul processus compose ce module et sa principale tâche est la gestion des communications par le port série.

En plus des processus, trois périphériques d'entrée et deux de sortie font aussi partie de l'architecture du simulateur de poste. L'unité de reconnaissance vocale de même que la souris 2D et le gant PowerGlove de Mattel permettent de commander le simulateur de postes et sont donc des périphériques d'entrée. Les écrans des stations de travail, qui affichent la simulation 3D, les photos et les clips vidéo, ainsi que les haut-parleurs

des stations, qui permettent l'audition des différents bruits et des messages d'erreur de manœuvres, sont les périphériques de sortie.

La figure 3-2 représente les processus, plutôt que les modules, composant le simulateur de postes en réalité virtuelle. Les périphériques sont aussi représentés sur cette figure. Les traits indiquent encore ici qu'il existe un moyen de communication entre ces divers processus.

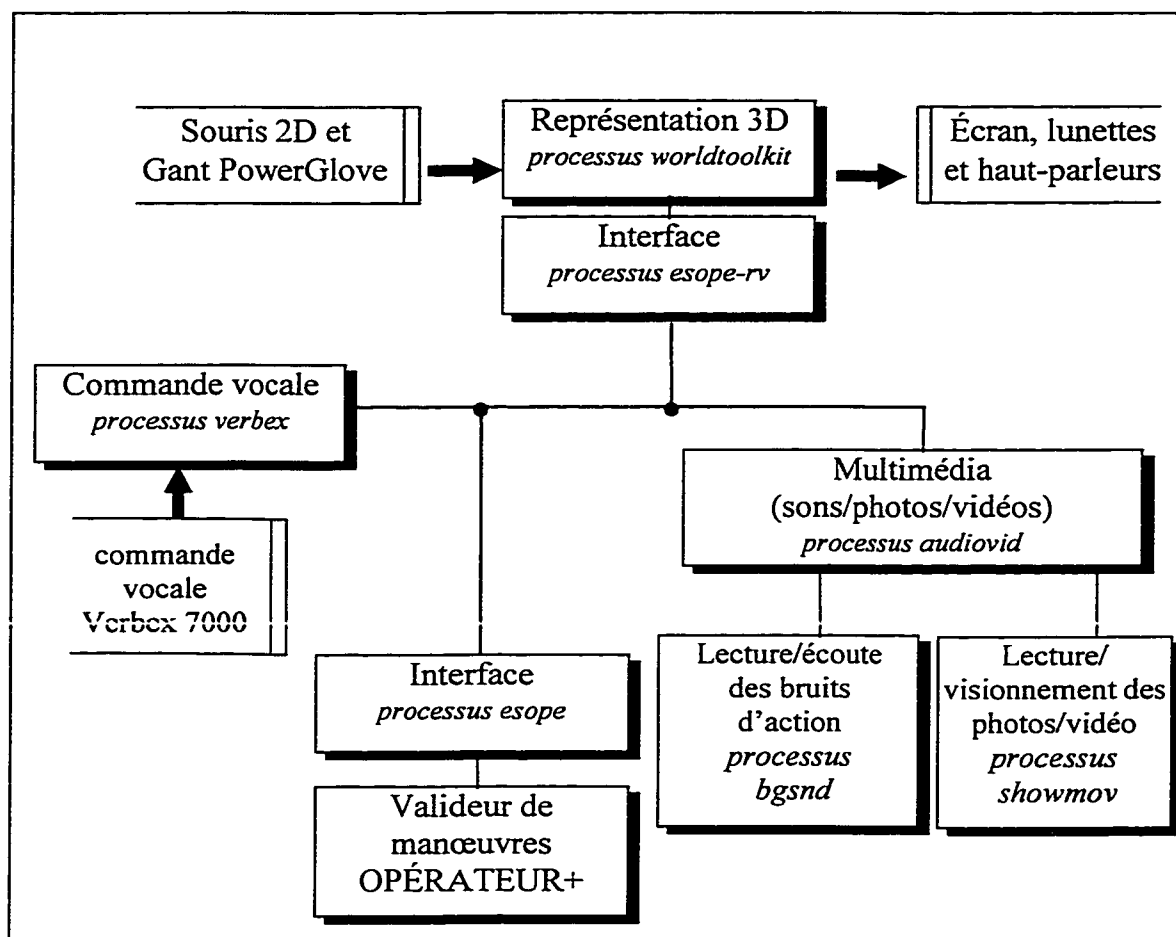


Figure 3-2 Processus du simulateur ESOPE-RV

3.4 Communications entre processus

Il existe plusieurs façons d'échanger de l'information entre des processus. Le système d'exploitation IRIX offre près d'une dizaine de moyens de communication différents lorsque deux processus ou plus désirent échanger (Stevens, 1990). Le choix d'un moyen plutôt qu'un autre dépend en partie de la relation existante entre les processus. On retrouve trois types de relation entre les processus sur les postes de travail de type

UNIX (Stevens, 1990): parent-enfant sur le même poste de travail, non-reliée sur le même poste de travail et non-reliée sur des postes de travail différents. Une relation parent-enfant entre deux processus indique qu'un processus a été créé par un autre processus à l'aide de la commande `fork` (Silverio et Ferguson, 1994). Cette commande permet de créer un nouveau processus qui est en fait une copie du processus parent. Cette copie est habituellement désignée comme étant l'enfant. Les deux processus se partagent la même mémoire physique, ce qui les oblige à être sur le même poste de travail. Les deux autres types représentent des processus sans lien de parenté (non-relié): s'ils s'exécutent sur le même poste de travail, les processus partagent la même mémoire physique. L'endroit où s'exécute les processus est important car certains moyens de communication fonctionnent seulement si les processus partagent la même mémoire physique.

Le moyen de communication sélectionné doit permettre aux huit processus s'exécutant sur trois (3) stations de travail SGI et sur deux (2) ordinateurs de type PC d'échanger de l'information, malgré les différents systèmes d'exploitation utilisés.

Le choix d'un moyen de communication doit aussi tenir compte de l'évolution future de la puissance de calcul des ordinateurs. Dans quelques années, il sera peut être possible d'exécuter tous les processus sur un poste de travail et d'obtenir malgré tout une simulation réaliste d'un poste en réalité virtuelle. Le moyen de communication doit donc permettre d'exécuter les processus sur des postes de travail séparés ou encore sur le même poste lorsque la puissance des ordinateur sera suffisante.

Le système d'exploitation des postes de travail, IRIX 5.2, est basé sur la version d'UNIX d'AT&T connue sous le nom *UNIX System V Release 4* (AT&T, 1990). Le système IRIX intègre cependant les fonctions de réseaux d'une autre version UNIX, celle de l'université de Berkeley, également connu sous le nom *BSD4.3* (Horspool, 1992). Les protocoles de communications les plus intéressants supportés par IRIX et permettant l'échange d'information sont les suivants (Stevens, 1990; Bassler, 1994):

- *Sémaphores*: des sémaphores en mémoire sont accessibles à deux processus. Ces sémaphores peuvent prendre deux valeurs différentes, ouvert ou fermé, et ne permettent donc pas l'envoi d'une chaîne de caractères. Les sémaphores sont employés pour synchroniser une tâche entre deux processus, comme l'écriture dans un fichier commun.
- *file de message*: une file de message est gérée par le système d'exploitation. L'information est échangée entre deux processus par l'envoi de messages. Les processus doivent être obligatoirement sur le même poste de travail.
- *mémoire partagée*: une région de mémoire est accessible par plus d'un processus. C'est habituellement la façon la plus rapide d'échanger de l'information car les données ne sont écrites qu'une seule fois. En effet, l'information est écrite une fois par l'émetteur puis le récepteur est en mesure de lire immédiatement cette information afin d'effectuer un traitement particulier. Pour les autres protocoles, l'information doit préalablement être copiée dans une variable interne avant d'être utilisée. Les processus doivent évidemment se trouver sur le même poste de travail.

- *sockets*: un canal, permettant d'échanger des données, est créé entre deux processus en utilisant l'adresse internet des stations exécutant les processus. Les processus peuvent se retrouver sur des postes différents et l'information est échangée sous la forme de messages.

Pour les communications entre les processus d'ESOPÉ-RV, le protocole le plus approprié est l'utilisation de sockets. Les sockets peuvent être utilisés entre des processus sans se soucier de l'endroit où s'exécutent ces processus. Ces derniers peuvent aussi être situés sur un ordinateur de type PC car les sockets sont disponibles sur la plupart des systèmes d'exploitation pour PC. Les sockets offrent la possibilité d'échanger des données dans les deux directions. Finalement, la gestion de plusieurs sockets reliés à un processus est facilitée par la présence de la fonction *select* (Silverio et Ferguson, 1994) permettant d'interroger l'ensemble des sockets en une seule étape. La présence d'informations en attente d'être lues est alors facilement détectée. Les sockets présentent cependant un désavantage: certains paramètres doivent être connus des deux processus reliés. Pour permettre aux processus de connaître ces paramètres, ces derniers sont souvent inscrits dans un fichier d'en-tête commun aux deux processus. Toutefois, cette approche est peu flexible car toute modification des paramètres exige une recompilation des deux processus. Une approche différente consiste à utiliser un fichier de configuration contenant ces paramètres. Ce fichier est lu au moment opportun par chacun des processus. Lorsque des modifications sont apportées aux paramètres, il est seulement nécessaire de

redémarrer les processus pour rendre les changements effectifs. Cette approche, beaucoup plus flexible, est celle adoptée pour le simulateur ESOPE-RV (voir la section 2.9).

L'établissement des différents liens par sockets se fait en deux étapes: un premier processus signale au système d'exploitation qu'il attend une connexion sur un port particulier (Stevens, 1990). Ce processus est considéré comme étant le serveur pour ce lien particulier seulement. Par la suite, un second processus cherche à établir la connexion avec le premier processus sur le même port. Ce second processus possède quant à lui le statut de client mais encore une fois, seulement pour ce lien. Le lien de communication est à ce moment établi et il devient alors possible d'envoyer des messages. La lecture et l'écriture de messages sur un socket se fait en appelant les fonctions `write` et `read` (Silverio et Ferguson, 1994). Afin de déterminer si un message attend d'être lu, la fonction `select` est appelée. Cette fonction retourne une valeur différente lorsqu'un message est en attente. Il suffit ensuite de lire le message avec la fonction `read`.

En ce qui concerne les communications entre le processus de représentation 3D (**worldtoolkit**) et son processus d'interface (**esope-rv**), une méthode plus rapide est cependant souhaitable. La technique utilisant la mémoire partagée est donc adoptée. Ce choix permet de maximiser l'utilisation du processeur de la station la plus puissante. Cette technique est aussi employée par le module multimédia pour les

mêmes besoins de performance. La création d'une portion de mémoire partagée est relativement facile. La mémoire partagée est créée à l'aide des fonctions `shmget` et `shmat` (Silverio et Ferguson, 1994). Les deux processus impliqués font appel à ces fonctions avec une clé commune. Cette clé, qui définit un endroit en mémoire, est conservée dans un fichier compilé avec les deux processus. Après l'appel de ces fonctions, chaque processus obtient un pointeur au même bloc en mémoire. Les deux processus peuvent ensuite écrire ou lire dans ce bloc. Les deux premiers caractères du bloc de mémoire sont utilisés comme un sémaphore. Ce sémaphore permet de synchroniser les processus afin d'éviter des situations où deux processus écrivent en même temps dans la mémoire partagée. Un processus écrit dans ce bloc seulement si le sémaphore a une valeur prédéterminée.

La figure 3-3 présente l'architecture finale du simulateur de poste. Les modules, les processus, les différents périphériques de même que les moyens de communications sont représentés. Cette figure présente la configuration de base du simulateur qui nécessite trois postes de travail et deux PC.

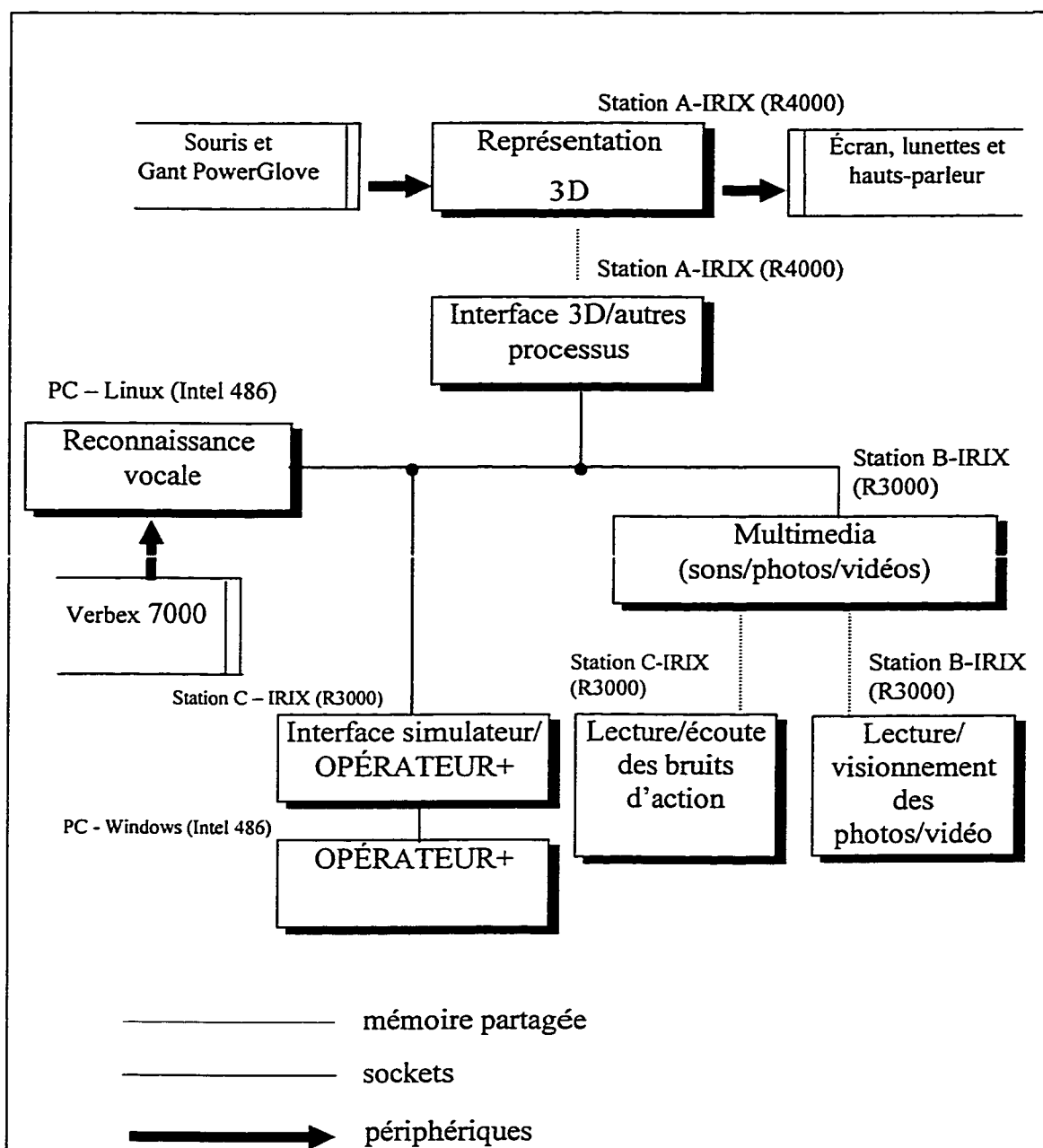


Figure 3-3 Architecture complète du simulateur 3D

3.5 Conclusion

L'architecture présentée permet d'exploiter les atouts des ordinateurs disponibles. De cette façon, le niveau de performance global est maximisé et peut être facilement amélioré en fournissant plus de puissance de traitement aux processus les plus exigeants. Cependant, la performance de la simulation n'est qu'un des aspects importants du simulateur de manœuvres en réalité virtuelle. L'intelligence de ce dernier réside dans sa capacité à valider les manœuvres de l'opérateur. Cette intelligence provient en fait du valideur de manœuvres d'OPÉRATEUR+, qui est intégré au simulateur ESOPE-RV. Le prochain chapitre décrit cette intégration.

4. Intégration d'OPÉRATEUR+

4.1 Introduction

L'objectif spécifique du présent projet est d'intégrer au simulateur ESOPE-RV les deux engins du valideur de manœuvres d'OPÉRATEUR+. Ces deux engins sont: l'engin de calcul d'écoulement de puissance et celui de validation de manœuvres par des règles. Cette intégration permet d'avoir accès rapidement à un engin de calcul d'écoulement de puissance flexible et robuste ainsi qu'à un mécanisme de validation de manœuvres facile d'utilisation. L'intégration des deux engins du valideur de manœuvres est décrite dans ce chapitre.

4.2 Exigences d'ESOPE-RV

Le simulateur ESOPE-RV se veut une extension de l'interface personne-système actuelle du valideur de manœuvres d'OPÉRATEUR+. Idéalement, l'utilisateur d'ESOPE-RV a accès aux mêmes fonctions du valideur de manœuvres que s'il utilisait ce dernier directement à partir de son interface 2D. Des contraintes de temps ont cependant limité le nombre de fonctions accessibles à partir d'ESOPE-RV. Par conséquent, seulement les fonctions nécessaires à la réalisation d'un plan de manœuvres type ont été réalisées. Les autres fonctions ne sont donc pas encore

disponibles directement à partir d'ESOPE-RV mais elles demeurent disponibles par l'interface 2D traditionnelle du valideur de manœuvres. L'ordinateur exécutant le valideur de manœuvres étant situé près du poste de travail affichant le poste électrique en réalité virtuelle, l'opérateur en formation peut exécuter certaines commandes inaccessibles directement par ESOPE-RV directement à partir de l'interface 2D du valideur de manœuvres.

Le plan de manœuvres type réalisable à l'aide d'ESOPE-RV consiste à isoler un disjoncteur de haute tension (voir l'annexe 1). Pour réaliser ce plan de manœuvres, il est primordial de pouvoir ouvrir et fermer un disjoncteur de même qu'un sectionneur. Il faut également modifier le mode de commande de ces appareils et être en mesure d'effectuer les vérifications les plus courantes, c'est à dire, vérifier l'état d'un appareil au panneau, vérifier l'état d'un appareil à son sémaphore, vérifier les conditions générales de même qu'inspecter la zone où les manœuvres sont effectuées. De plus, après la réalisation d'une manœuvre, il est nécessaire de noter le moment où cette manœuvre a été effectuée. Ces opérations devront donc être réalisables à partir de l'interface d'ESOPE-RV.

La représentation 3D de l'installation est construite à partir des données du schéma unifilaire chargé dans le valideur de manœuvres. Il est alors essentiel de pouvoir consulter, à partir d'ESOPE-RV, la valeur des paramètres d'un appareil du schéma, les connexions électriques entre chaque appareil de même que la position des appareils à

l'écran. Les paramètres d'un appareil (type d'appareil, tension nominale, etc.) permettent de choisir l'objet 3D représentant le mieux cet appareil. Les connexions électriques sont employées quant à eux pour positionner les appareils les uns par rapport aux autres dans l'environnement virtuel. Finalement, la position à l'écran d'un appareil est parfois utile pour déterminer la configuration de barres du poste (*double bus double breaker*, etc.), utilisée par l'algorithme de conversion de schéma afin de positionner de façon réaliste les appareils dans le poste virtuel.

Pendant l'exécution d'un plan de manœuvres par un opérateur en formation, le valideur de manœuvres remplit deux rôles distincts. Le premier rôle du valideur de manœuvres est d'approuver les manœuvres effectuées à partir de l'interface 3D du simulateur ESOPE-RV. Lorsqu'une manœuvre est effectuée par l'utilisateur, ESOPE-RV doit consulter le valideur de manœuvres pour déterminer si la manœuvre est valide ou non. Le valideur de manœuvres analyse la manœuvre à l'aide des règles d'exploitation internes et donne ensuite son verdict. Une manœuvre refusée est immédiatement signalée à l'utilisateur par le biais d'un message auditif alors qu'une manœuvre acceptée modifie habituellement l'apparence du poste dans la simulation 3D. Le second rôle du valideur de manœuvres est de calculer l'écoulement de puissance pour le schéma unifilaire. Ce calcul permet de déterminer les valeurs de tension, de courant et de puissance pour chacun des appareils du schéma. Ces valeurs sont ensuite affichées sur des cadrans, situés sur les panneaux de la salle de commande virtuelle.

4.3 Contraintes de réalisation

L'intégration au simulateur ESOPE-RV du valideur de manœuvres nécessite qu'une série de modifications soient effectuées à ce dernier. Le valideur de manœuvres est un produit présentement commercialisé par Hydro-Québec avec un plan de développement chargé et des échéanciers serrés. Un accès illimité à tous les fichiers de sources du valideur de manœuvres par l'équipe d'ESOPE-RV n'étant pas possible, les modifications nécessaires sont effectuées par l'équipe de développement d'Hydro-Québec. Pour cette raison, les modifications apportées au valideur de manœuvres sont limitées au strict car le temps disponible à l'équipe de développement d'Hydro-Québec pour effectuer ces modifications est très court.

L'engin de validation de manœuvres est un des deux engins du valideur à être intégré. L'engin de validation est actuellement accessible de l'extérieur de l'application. En effet, il est possible de soumettre des commandes à l'engin de validation de manœuvres à partir d'un autre ordinateur à l'aide d'un logiciel de type Telnet. Il suffit d'établir une connexion sur le port de communication 80 de l'ordinateur qui exécute le valideur de manœuvres puis d'inscrire les commandes à soumettre dans la fenêtre du logiciel Telnet. Les commandes, qui doivent respecter une certaine syntaxe, permettent d'accéder aux paramètres des appareils, aux vérifications et aux connexions électriques des appareils. Le protocole de communication TCP/IP (Comer, 1991) doit cependant être installé sur les deux ordinateurs impliqués. Il est également possible d'accéder au

port de communication 80 à partir d'une application (autre que Telnet) en établissant un lien par socket entre l'application et ce même port. L'application peut d'abord soumettre une commande à l'engin de validation de manœuvres en utilisant la fonction d'écriture sur un socket, la fonction `write`, sur ce lien. Le résultat de la commande peut ensuite être lu à partir de ce même lien, en utilisant maintenant la fonction de lecture d'un socket, la fonction `read`. Cette méthode est la seule qui permet d'accéder au valideur de manœuvres à partir d'une application s'exécutant sur poste de travail de type UNIX.

4.4 *Processus d'interface ESOPE*

L'intégration du valideur de manœuvres d'OPÉRATEUR+ se fait par l'utilisation d'un processus supplémentaire. Ce processus, appelé **esope**, sert de pont entre les autres processus d'ESOPE-RV et le valideur de manœuvres, gérant ainsi l'ensemble des communications entre tous ces processus. Comme l'indique la figure 4-1, le processus **esope** est le seul point d'entrée vers le valideur de manœuvres.

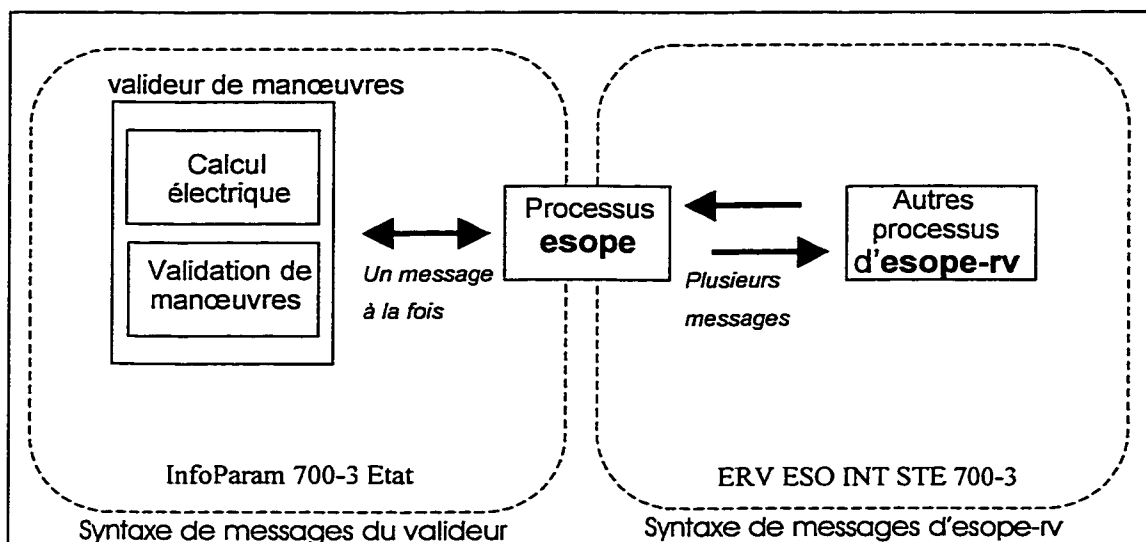


Figure 4-1 Processus esope

Ce processus supplémentaire permet, entre autres, de limiter l'impact des modifications apportées au valideur de manœuvres sur les autres processus d'ESOPE-RV. Le valideur de manœuvres est un produit commercial qui est en constante évolution afin de le perfectionner et de l'améliorer. Un changement dans la syntaxe d'un paramètre d'un appareil nécessiterait normalement des modifications dans chacun des processus d'ESOPE-RV. En permettant seulement au processus ESOPE d'avoir des échanges directs avec le valideur de manœuvres, les modifications, causées par des changements apportés au valideur de manœuvres, sont limitées à un seul processus.

L'utilisation du processus **esope** en tant que seul point d'accès au valideur de manœuvres, pour les autres processus d'ESOPE-RV, présente également d'autres

avantages. Par exemple, il est possible de soumettre les demandes au valideur de manœuvres par ordre de priorité. Les demandes urgentes, suite à une manœuvre par exemple, sont soumises immédiatement alors que les moins importantes, comme la mise à jour des cadrans de la salle de commande, sont traitées ultérieurement. Le processus **esope** offre aussi la possibilité d'effectuer certains traitements reliés à la gestion des commandes uniquement à l'intérieur du processus **esope** plutôt que dans l'ensemble des processus. Certains processus, en particulier celui responsable de la représentation 3D du poste, doivent limiter les traitements effectués qui ne sont pas reliés directement à leur responsabilité première, afin de ne pas diminuer les performances du simulateur ESOPE-RV.

Le processus **esope** centralise également le traitement des erreurs provenant du valideur de manœuvres à l'intérieur d'un seul processus, ce qui a pour effet de simplifier la gestion des erreurs.

4.5 Modifications au valideur de manœuvres

Les premières modifications nécessaires à l'intégration du valideur de manœuvres au projet ESOPE-RV se situent à l'intérieur du valideur de manœuvres. Ces modifications ont été effectuées par Hydro-Québec après avoir consulté l'équipe d'ESOPE-RV. Elles avaient pour objet de permettre l'intégration de leur logiciel. Toutefois, avant de

présenter les modifications nécessaires, il est essentiel de préciser le rôle du valideur de manœuvres dans la réalisation d'un plan de manœuvres.

Lors du démarrage d'ESOPE-RV, le valideur de manœuvres est consulté pour obtenir l'information nécessaire à la construction en 3D de l'installation: le nom, le type, l'état ainsi que les niveaux de tension pour chaque appareil sont alors exigés. La position des appareils dans le schéma unifilaire, parfois utilisée lors de la détermination de la configuration de barres d'un poste, est également exigée. De plus, les liaisons électriques entre chaque appareil de même que la position physique initiale de l'opérateur sont obtenus au démarrage d'ESOPE-RV. C'est à ce moment que le trafic de messages entre le valideur de manœuvres et le processus **esope** est le plus élevé.

Après la construction du poste en réalité virtuelle, les échanges entre le valideur de manœuvres et **esope** diminuent en fréquence: elles se produisent uniquement suite à une action de l'utilisateur dans l'installation virtuelle. Il faut bien entendu que l'action nécessite une validation de la part du valideur de manœuvres. En effet, quelques actions de l'opérateur n'exigent pas de validation par le valideur de manœuvres. Par exemple, se déplacer dans la salle de commande ou à l'extérieur n'implique pas de validation par le valideur de manœuvres alors que le changement d'un paramètre d'un appareil (l'état, le mode de commande, etc.) est toujours validé par le valideur de manœuvres. Lorsqu'une action exige d'être validée, elle est premièrement signalée au valideur de manœuvres. Après quelques instants, le valideur transmet son verdict à

esope. Lorsque l'action est accepté par le valideur de manœuvres, le changement est intégré au poste virtuel tandis qu'une action refusée provoque l'écoute d'un message audio indiquant la nature de l'erreur.

Certaines actions, comme le changement de la position physique de l'opérateur, n'ont pas à être acceptées par le valideur de manœuvres. Ce dernier est tout de même avisé de la nouvelle position afin qu'il puisse mettre à jour ses structures de données internes. Finalement, le valideur n'est nullement impliqué lorsque l'opérateur consulte une photo ou un vidéo ou encore lorsqu'il demande de l'aide par commande vocale.

Les commandes existantes de l'engin de validation de manœuvres permettent de lire les paramètres des appareils (comme l'État, le Mode de commande, etc.) et les connexions électriques ainsi que de consulter les vérifications réalisées antérieurement sur chaque appareil. Cependant, le changement d'un paramètre et sa validation se font seulement à partir de l'interface 2D du valideur de manœuvres. Des modifications ont donc été apportées afin de permettre de réaliser des manœuvres à partir du simulateur ESOPE-RV.

La gestion des erreurs de manœuvres par le valideur de manœuvres doit aussi être modifiée. En effet, les messages d'erreur, suite à une manœuvre interdite, sont présentement affichés à l'écran dans un dialogue sous une forme textuelle. Ces messages requièrent de la part de l'utilisateur qu'il ferme le dialogue en cliquant sur le

bouton OK. Comme l'approche sélectionnée par ESOPE-RV pour signaler les erreurs est d'utiliser des messages audio expliquant l'erreur, il n'est pas nécessaire d'avoir un tel dialogue d'erreur à l'écran du valideur de manœuvres. Ce dialogue devra donc être inhibé lorsque la manœuvre provient d'ESOPE-RV. Le valideur de manœuvres doit seulement indiquer au processus **esope** qu'une erreur a été commise et permettre à ce dernier de déterminer la nature de l'erreur. Les erreurs d'une même nature sont associées à un code d'erreur utilisé pour identifier le message audio correspondant à l'erreur et éventuellement le faire entendre à l'opérateur.

4.6 Moyens de communication

Les communications entre le valideur de manœuvres et le processus **esope** se font par l'entremise des sockets sous le protocole TCP/IP (Comer, 1991) car le valideur supporte déjà ce type de communication. Un lien de communication par sockets existe également entre le processus **esope** et les processus suivants d'ESOPE-RV: **verbex**, **esope-rv** et **audiovidéo**. Au total, le processus **esope** doit établir quatre liens par sockets et effectuer la gestion des messages provenant de chacun de ces liens.

Lors de l'établissement d'un lien par socket, un des deux processus agit comme le serveur alors que l'autre est le client. Le processus **esope** prend le rôle d'un serveur pour les liens avec **esope-rv**, avec **audiovidéo** ainsi qu'avec le valideur de manœuvres. En ce qui concerne le lien avec le processus **verbex**, ce dernier est le serveur alors que

le processus **esope** est le client. Le rôle identifie l'instigateur du lien entre deux processus (voir chapitre 3). Lorsque tous les liens sont établis, la communication par messages peut débuter. Le format des messages de même que l'interprétation des messages par les processus sont discutés dans les sections 4.7 et 4.8.

4.7 *Format des messages*

Un message contient une information devant être transmise d'un processus à un autre. En plus de l'information proprement dite, le message contient d'autres éléments permettant de faciliter l'envoi et le traitement du message tel que, le destinataire du message ainsi que l'émetteur. Ces différents éléments ne sont pas placés au hasard à l'intérieur du message. En effet, il existe une syntaxe particulière devant être respectée par chaque message. La syntaxe des messages a été établie par les membres du projet ESOPE-RV de façon à ce que les messages soient lisibles, courts et extensibles. Les messages sont constitués d'éléments, qui sont des chaînes de textes ayant au moins un caractère. Les éléments d'un message sont positionnés un à la suite de l'autre et sont séparés par un espace.

Les éléments respectent toujours l'ordre suivant (de gauche à droite): le destinataire, l'émetteur, le type de message puis la commande proprement dite. La commande, qui indique la manœuvre qui a été réalisée ou l'information qu'on désire obtenir, est

composée de plusieurs sous-éléments également séparés par des espaces. Par exemple, le message de l'exemple 1 est reçu, suite à l'ouverture du disjoncteur 700-3 :

EXEMPLE 1

ERV ESO MAN OPN BRK 700-3

1 2 3 4 5 6

1. émetteur du message
2. destinataire du message
3. le type de message. Ici, le message annonce qu'une manœuvre a été réalisée.
- 4,5 et 6. la commande avec les sous-éléments suivants:
 4. la manœuvre, on a ouvert un appareil (opn = open)
 5. l'appareil manoeuvré, un disjoncteur (brk = circuit breaker)
 6. le nom de l'appareil (700-3)

Le premier élément d'un message est l'émetteur du message. Dans l'exemple présenté, le message provient du processus **esope-rv** (abrégé à ERV), l'interface du processus responsable de la représentation 3D du poste. Le second élément est le destinataire du message, soit le processus **esope** (abrégé à ESO) qui agit comme interface entre le valideur de manœuvres et les autres processus du simulateur ESOPÉ-RV. L'élément présenté par la suite est le type de message. Les choix possibles sont: manœuvre, interrogation, navigation ou erreur (abrégé à INT, MAN, NAV et ERR

respectivement). Une manœuvre indique qu'un paramètre d'un appareil a été modifié par l'opérateur, une interrogation sert à obtenir une information sur un appareil (ex: le niveau de tension), une navigation signale le changement de la position physique de l'opérateur alors que le type « erreur » signifie que l'opérateur a effectué une erreur en ne respectant pas une règle d'exploitation ou une méthode de travail. Les autres éléments d'un message, leur nombre variant habituellement de un à quatre, représentent la commande ou l'information à échanger. Le nombre de sous-éléments dans la commande dépend du type du message, indiqué par le troisième élément.

4.7.1 Messages de type « manœuvre »

Le tableau 4-1 contient les messages pouvant être envoyés au processus **esope** suite à une manœuvre (type manœuvre). Le tableau ne contient cependant pas les deux premiers éléments des messages, soit l'émetteur et le destinataire. Pour ces messages destinés au processus **esope**, l'émetteur est soit VBX (processus **verbex**, la commande vocale) ou ERV (processus **esope-rv**, l'interface pour la représentation 3D) tandis que le destinataire est exclusivement le processus **esope** ayant l'abréviation ESO.

Tableau 4-1 Messages de type « manœuvre »

Élément #3	Élément #4	Élément #5	Élément #6	Description de la manœuvre
MAN	CLO	BRK ou SWI	Nom de l'appareil	Fermer un sectionneur(SWI) ou un disjoncteur(BRK)
MAN	INS			Inspecter la zone concernée par les manœuvres
MAN	LOC	BRK ou SWI	Nom de l'appareil	Mettre à Local le mode télé d'un sectionneur(SWI) ou d'un disjoncteur(BRK)
MAN	OPN	BRK ou SWI	Nom de l'appareil	Ouvrir un sectionneur(SWI) ou un disjoncteur(BRK)
MAN	TEL			Activer le mode télémanœuvre, permettant de faire une manœuvre en étant à l'équipement, plutôt qu'à la salle de commande.
MAN	TIM			Note l'heure de la dernière manœuvre
MAN	TLE	BRK ou SWI	Nom de l'appareil	Mettre à Télé le mode télé d'un sectionneur(SWI) ou d'un disjoncteur(BRK)
MAN	VFY			Vérifier les conditions d'exploitation de l'installation
MAN	VPA	BRK ou SWI	Nom de l'appareil	Vérifier au panneau l'état d'un disjoncteur (BRK) ou d'un sectionneur (SWI)
MAN	VSE	BRK ou SWI	Nom de l'appareil	Vérifier aux sémaphores l'état d'un disjoncteur (BRK) ou d'un sectionneur (SWI)

Lorsqu'une manœuvre est signalée au processus **esope**, ce dernier consulte le valideur de manœuvres afin de déterminer si la manœuvre est valide. Les modalités de consultation du valideur sont exposées prochainement. Si la consultation renvoi un résultat affirmatif au processus **esope**, ce dernier envoie un message au processus

responsable de la représentation 3D, **worldtoolkit**, en passant préalablement par le processus **esope-rv**. Ce message permet ensuite de modifier la représentation du poste à l'écran de façon à intégrer la manœuvre acceptée. Pour le message de l'exemple 1, la syntaxe du message de retour envoyé par le processus **esope** est la suivante:

MESSAGE DE RETOUR DE L'EXEMPLE 1

ESO ERV MAN OPN BRK 700-3

Le message est identique à celui envoyé par le processus **esope-rv** (ERV), reçu initialement par le processus **esope**, à l'exception de l'émetteur et du destinataire qui ont été inversés.

4.7.2 Messages de type « interrogation »

La syntaxe du deuxième type de message, les messages de type interrogation, est légèrement différente des messages de type manœuvre. L'exemple 2 présente un message qui permet d'obtenir la tension de l'appareil spécifié, la ligne L6.

EXEMPLE 2

ERV	ESO	INT	VLT	L6
1	2	3	4	5

1. émetteur du message
2. destinataire du message
3. le type de message. Le message est une interrogation
- 4 et 5 la commande avec les sous-éléments suivants:
 4. le paramètre qu'on désire connaître, la tension (VLT=voltage).
 5. l'appareil en question, la ligne L6.

On remarque que pour un message de type interrogation, la signification des sous-éléments de la commande est différente. Dans le dernier exemple, les items 4 et 5 n'ont plus la même signification tandis que l'item 6 n'est plus présent.

Le prochain tableau contient tous les messages de type interrogation. Il est ainsi possible d'obtenir de l'information sur un schéma unifilaire, sur un appareil en particulier ou sur un type d'appareil. Les types d'appareils supportés par le processus **esope** sont: le transformateur (abrégié par l'élément XFO), le sectionneur (SWI), le disjoncteur (BRK), la ligne (LIG) et la barre (BAR).

Tableau 4-2 Messages de type « interrogation »

Élément #3	Élément #4	Élément #5	Élément #6	Description de l'interrogation
INT	CUR	Nom de l'appareil		Obtenir le courant dans un appareil
INT	CUR	BAR, BRK, LIG, SWI ou XFO	CUR=XX	Obtenir une liste d'appareils du type spécifié dont le courant est tel que celui spécifié.

Élément #3	Élément #4	Élément #5	Élément #6	Description de l'interrogation
INT	DPU	Nom de l'appareil		Obtenir la direction arbitraire de la puissance(Entrée ou Sortie) pour une ligne
INT	DPU	LIG	DPU=Entrée ou Sortie	Obtenir une liste d'appareils du type spécifié dont la direction arbitraire de la puissance est telle que celle spécifiée.
INT	MLO	Nom de l'appareil		Obtenir le mode local d'un appareil
INT	MLO	BRK ou SWI	MLO=Distance ou Directe	Obtenir une liste d'appareils du type spécifié dont le mode local est tel que celui spécifié.
INT	MTE	Nom de l'appareil		Obtenir le mode télé d'un appareil
INT	MTE	BRK ou LIG	MTE=Télé ou Local	Obtenir une liste d'appareils du type spécifié dont le mode télé est tel que celui spécifié.
INT	NBR	BAR, BRK, LIG, SWI ou XFO		Obtenir le nombre d'appareil du type spécifié.
INT	PAC	Nom de l'appareil		Obtenir la puissance active d'un appareil
INT	PAC	BAR, BRK, LIG, SWI ou XFO	PAC=XX	Obtenir une liste d'appareils du type spécifié dont la tension est telle que celle spécifiée.
INT	PRE	Nom de l'appareil		Obtenir la puissance réelle (MW) d'un appareil
INT	PRE	BAR, BRK, LIG, SWI ou XFO	PRE=XX	Obtenir une liste d'appareils du type spécifié dont la puissance réelle (MW) est telle que celle spécifiée.
INT	STE	Nom du disjoncteur ou sectionneur		Obtenir l'état d'un appareil. L'appareil doit être un sectionneur ou un disjoncteur

Élément #3	Élément #4	Élément #5	Élément #6	Description de l'interrogation
INT	STE	SWI ou BRK	STE=Ouvert ou Fermé	Obtenir une liste d'appareils du type spécifié dont l'état est tel que celui spécifié.
INT	VL1	Nom du transformateur		Obtenir la tension nominale du coté primaire du transformateur
INT	VL1	XFO	VL1=XX	Obtenir une liste de transformateurs dont la tension nominale primaire est telle que celle spécifiée.
INT	VL2	Nom de l'appareil		Obtenir la tension nominale secondaire pour un transformateur
INT	VL2	XFO	VL2=XX	Obtenir une liste d'appareils du type spécifié dont la tension nominale secondaire est telle que celle spécifiée.
INT	VL3	Nom de l'appareil		Obtenir la tension nominale tertiaire pour un transformateur
INT	VL3	XFO	VL3=XX	Obtenir une liste d'appareils du type spécifié dont la tension nominale tertiaire est telle que celle spécifiée.
INT	VLT	Nom de l'appareil		Obtenir la tension d'un appareil
INT	VLT	BAR, SWI, BRK, TXF ou LIG	VLT=XX	Obtenir une liste d'appareils du type spécifié dont la tension est telle que celle spécifiée.
INT	XST	Nom de l'appareil		Obtenir la confirmation qu'un appareil existe

Les messages ayant un sixième élément sont dit conditionnels. Ces messages retournent au processus émetteur une liste d'appareils dont la condition, le sixième élément, est respectée. Par exemple, un message ayant comme condition VLT=120

permet d'obtenir une liste d'appareils dont la tension est de 120 kV. Cette liste peut évidemment être vide.

Contrairement aux messages de type manœuvre, la syntaxe des messages de retour pour les messages de type interrogation est différente du message initial. Pour l'exemple donné précédemment, le message retourné au processus ESOPE-RV par **esope** serait le suivant:

MESSAGE DE RETOUR DE L'EXEMPLE 2

```
ERV  ESO  INT  L6   VLT  735.00
```

Le message est sensiblement le même que celui reçu initialement par ESOPE-RV. Les items 4 et 5 sont cependant inversés et la valeur du paramètre désiré est ajoutée à la fin. L'inversion a pour but de faciliter l'analyse du message par le processus qui le reçoit. À la réception du message, ce dernier est analysé de gauche à droite. L'inversion permet de lire l'identification de l'appareil (L6 dans le dernier message) avant le paramètre et sa valeur (VLT et 735.00 respectivement). L'identification d'un l'appareil étant unique dans un schéma, on peut utiliser l'identification de l'appareil comme une clé ou un index, afin d'accéder au bon endroit dans une structure de données en mémoire ou sur le disque rigide.

Les messages qui retournent une liste d'appareils ont cependant une syntaxe différente. En effet, le nombre d'éléments dans le message de retour est alors plus élevé car une liste d'appareils est annexée au message. Il existe deux cas où le message de type interrogation retourne une liste: lorsque le nombre d'appareils est interrogé (NBR) de même que pour les messages conditionnels. Par exemple, voici un message reçu par **esope** en provenance du processus ESOPE-RV et permettant d'obtenir le nombre de lignes dans le schéma unifilaire:

EXEMPLE 3

ERV ESO INT NBR LIG

1 2 3 4 5

1. émetteur du message
2. destinataire du message
3. le type de message. Le message est une interrogation
- 4 et 5 la commande avec les sous-éléments suivants:
 4. le paramètre qu'on désire connaître, le nombre d'appareils du type spécifié
 5. le type d'appareil, les lignes du schéma.

Le message retourné à ESOPE-RV est alors assez différent par rapport au message original :

MESSAGE DE RETOUR DE L'EXEMPLE 3

ESO	ERV	INT	LIG	NBR	4	NOCOND	L1 L2 L7 L8
1	2	3	4	5	6	7	8

Les éléments 4 et 5 du message original sont inversés, comme pour les messages d'interrogation qui ne retournent pas une liste d'appareils. Le résultat de la demande d'interrogation, soit le nombre de lignes, est le sixième élément du message de retour. Toutefois, des éléments se sont ajoutés après l'élément 6. En effet, l'élément 7 indique que le message n'est pas du type conditionnel (de l'anglais, « no conditions », qui signifie *pas de conditions*) alors qu'une liste des appareils du schéma du type demandé est retournée (les lignes L1, L2, L7 et L8) à la fin du message.

Pour ce qui est du deuxième cas d'exception, les messages conditionnels, le format du message retourné par **esope** est sensiblement le même que pour le premier cas. En effet, la seule différence est l'élément 7, qui est remplacé par la condition spécifiée dans le message d'origine. Voici un message conditionnel qui permet d'obtenir la liste des lignes dont la tension nominale est de 735kV:

EXEMPLE 4

ERV ESO INT VLT LIG VLT=735.00

1 2 3 4 5 6

1. l'émetteur du message
2. le destinataire du message
3. le type de message. Le message est une interrogation
- 4 et 5 la commande avec les sous-éléments suivants:
 4. le paramètre qu'on désire connaître, le nombre d'appareils du type spécifié
 5. le type d'appareil, les lignes du schéma
 6. la condition, la tension doit être de 735.00 kV

Le message retourné par **esope** serait donc le suivant:

MESSAGE DE RETOUR DE L'EXEMPLE 4

ESO ERV INT LIG NBR 2 VLT=735.00 L8 L6

1 2 3 4 5 6 7 8

Dans ce message, le nombre d'appareils au total est retourné, comme dans l'exemple 3. L'élément 7 est cependant différent dans cet exemple car seulement les lignes qui répondent à la condition spécifiée sont annexées à la fin du message alors que pour l'exemple 3, toutes les lignes étaient annexées.

4.7.3 Messages de type « navigation »

Le troisième type de messages, les messages de type navigation suite à un changement de la position physique de l'opérateur, ont eux aussi une syntaxe particulière. Par exemple, un déplacement vers la salle de commande se fait à l'aide du message suivant, envoyé au processus **esope**:

EXEMPLE 5

ERV	ESO	NAV	GTO	CTR
1	2	3	4	5

1. l'émetteur du message
2. le destinataire du message
3. le type de message. Le message est un changement de position de l'opérateur.
- 4 et 5 la commande avec les sous-éléments suivants:
 4. le type de déplacement effectué
 5. la nouvelle position de l'opérateur

Ce type de message contient toujours deux éléments suite à la catégorie de message. Le quatrième élément est toujours GTO (de l'anglais "goto") et signifie que l'opérateur a changé physiquement de position. Quant au dernier élément, il indique la nouvelle

position de l'opérateur. Dans l'exemple, l'opérateur se situe maintenant à la salle de commande. Voici les messages de type navigation pouvant être reçu par **esope**:

Tableau 4-3 Messages de type « navigation »

Élément #3	Élément #4	Élément #5	Élément #6	Description du déplacement
NAV	GTO	YRD		L'opérateur est maintenant à l'équipement, à l'extérieur du poste
NAV	GTO	CTR		L'opérateur est maintenant dans la salle de commande du poste

Une différence importante entre les messages de navigation et les autres types de message se situe au niveau du message retourné au processus émetteur. En effet, les messages de type navigation ne provoquent pas de message de confirmation vers le processus émetteur. Ceci provient du fait que les changements de position de l'opérateur ne demandent aucune validation de la part du valideur de manœuvres. Ce dernier reçoit le message uniquement pour être informé de la modification de la position et pour mettre à jour sa variable interne contenant la position courante de l'opérateur. Ceci signifie qu'au moment où le processus **esope** reçoit l'avis de changement de position, la position de l'opérateur dans la représentation en réalité virtuelle du poste a déjà été modifiée à sa nouvelle valeur.

4.7.4 Messages de type « erreur »

En ce qui concerne le dernier type de messages, les messages de type erreur, on peut également constater des changements de syntaxe. Par exemple, voici un message envoyé par **esope** indiquant une erreur:

EXEMPLE 6

```
ESO  A_V  ERR  1001
```

```
1      2      3      4
```

1. l'émetteur du message
2. le destinataire du message
3. le type de message. Le message indique qu'une erreur s'est produite.
- 4 la commande avec le sous-élément suivant:
 4. le code de l'erreur, 1001 dans cet exemple.

Dans l'exemple 6, le nombre de sous-éléments de la commande a diminué: il n'en reste qu'un seul, qui indique le code de l'erreur produite. Les messages de type « erreur » ne sont jamais reçus par le processus **esope** car c'est plutôt ce dernier qui les envoie aux autres processus. En effet, la détection des erreurs est réalisée à l'intérieur du processus **esope**. Lorsqu'une erreur est détectée par le processus **esope**, ce dernier envoie un message signalant l'erreur au processus **audiovidéo** (A_V), qui fera

entendre le message audio associé à ce code d'erreur. Le tableau 4-4 présente le format des messages d'erreur, envoyé par **esope** (ESO) au processus **audiovidéo** (A_V).

Tableau 4-4 Messages de type « erreur »

Elément #3	Elément #4	Elément #5	Elément #6	Description de l'erreur
ERR	Code d'erreur			L'erreur associée à ce code d'erreur s'est produite.

Les différents codes d'erreur possibles sont présentés à la section 4.8.6 de ce chapitre.

Il est important de noter que la syntaxe des messages présentées dans ce chapitre ne sert pas au valideur de manœuvres. En effet, ce dernier possède sa propre syntaxe imposée par l'engin de validation de manœuvres. Le processus **esope** doit donc effectuer la conversion entre la syntaxe utilisée par les différents processus d'ESOPE-RV et celle du valideur de manœuvres. La prochaine section traite, entre autres, de cette conversion.

4.8 Gestion des messages par ESOPE

Les différentes syntaxes de messages possibles augmentent la probabilité de recevoir, par le processus **esope**, un message mal formulé, surtout pendant la période de développement et de mise au point du simulateur ESOPE-RV. Afin d'éviter que des messages illogiques soient soumis par **esope** au valideur de manœuvres, le processus

esope s'assure de la validité de tous les messages qu'il reçoit des autres processus d'ESOPE-RV.

En plus de valider chaque message reçu, le processus **esope** gère également le flux de messages dirigé vers lui par les autres processus, afin d'être soumis au valideur de manœuvres. Cette gestion est nécessaire puisque le valideur de manœuvres ne peut répondre à plusieurs demandes en parallèle. En effet, avant de soumettre au valideur de manœuvres un autre message, le traitement du message courant doit obligatoirement être terminé. Pendant le traitement d'un message par le valideur de manœuvres, les messages reçus par le processus **esope** sont emmagasinés dans une file d'attente.

La gestion des messages destinées au valideur de manœuvres par le processus **esope** se fait essentiellement en huit étapes, qui sont continuellement répétées. Ces étapes sont:

1. Lecture du message destiné au valideur de manœuvres;
2. Validation du message;
3. Conversion du message en une commande pour le valideur de manœuvres;
4. Envoi de la commande au valideur de manœuvres;
5. Lecture du résultat provenant du valideur de manœuvres;
6. Validation du résultat;
7. Conversion du résultat en un message;
8. Envoi du message avec le résultat au processus approprié.

Le diagramme de la figure 4-2 présente la façon dont sont traités les messages destinés au processus **esope**. De plus, les huit étapes énumérées précédemment sont clairement indiquées sur le diagramme. Ces étapes sont ensuite abordées dans les prochaines sections.

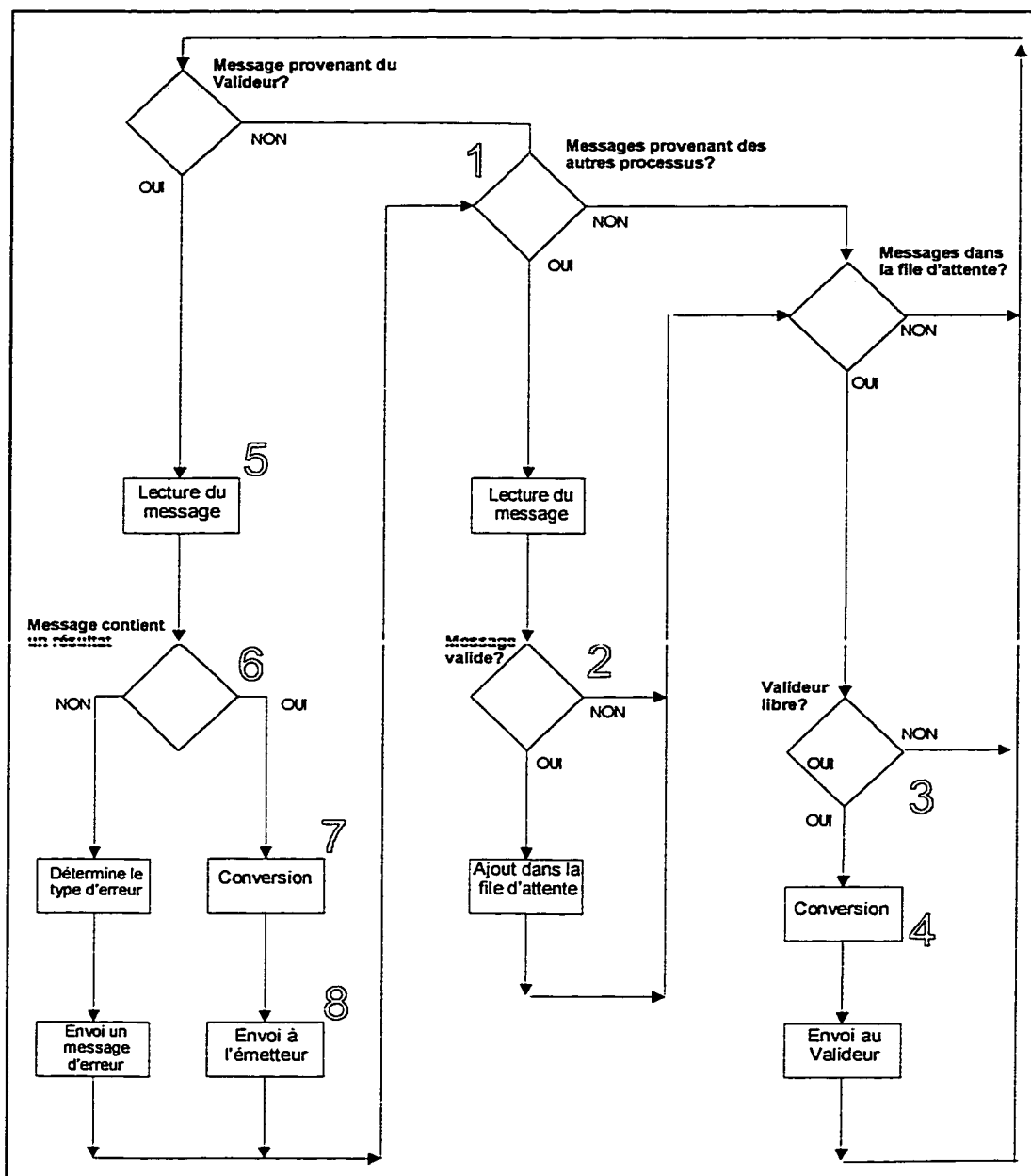


Figure 4-2 Gestion des messages par esope

4.8.1 Lecture des messages destinés au valideur

La première étape débute après avoir initialisé tous les liens avec les autres processus. Le processus **esope** vérifie la présence d'un message sur chacun des quatre liens en utilisant continuellement la fonction `select`. Lorsqu'un message est détecté par cette fonction, chaque lien est testé afin de déterminer le lien sur lequel se situe le message. Lorsqu'un test se révèle positif, le message, composé d'une chaîne de caractères, est immédiatement lu à l'aide de la fonction `read`.

Les liens sont testés par le processus **esope** selon un ordre particulier: le lien vers le valideur de manœuvres est toujours le premier, suivi des liens vers **verbex**, **esope-rv** puis **audiovidéo**. Les messages provenant du valideur de manœuvres sont lus les premiers afin de libérer ce dernier le plus rapidement possible. En effet, un message provenant du valideur contient la réponse au dernier message lui ayant été. Ce dernier ne pouvant pas traiter plus d'un message à la fois, il est avantageux de lire rapidement le résultat du dernier message envoyé au valideur de manœuvres afin de lui soumettre le plus rapidement possible un nouveau message.

Lorsque la lecture des messages provenant du valideur de manœuvres est terminée, les messages en provenance des autres processus sont lus par le processus **esope**. Chaque message lu est copié dans une variable de type MESSAGE dont la mémoire est allouée dynamiquement. Cette variable est une structure de données dont la définition est en

annexe de ce document (voir l'annexe 4). Les messages ne sont pas traités immédiatement par le processus **esope**: la variable contenant le message est plutôt ajoutée à la fin d'une file d'attente interne.

4.8.2 Validation du message

La seconde étape consiste à s'assurer que la syntaxe du message qui sera éventuellement soumis au valideur de manœuvres est valide. Après avoir sondé chacun des liens pour détecter des messages, le premier message de la file d'attente interne est validé. La validation s'effectue en analysant chacun des éléments du messages et en les comparant avec une table en mémoire contenant la syntaxe admissible. L'orthographe des éléments, ainsi que leur position dans le message, sont à ce moment validés. Cependant, cette table ne permet pas de valider en une seule opération l'ensemble des éléments d'un message. En effet, l'identification de l'élément et, dans le cas des messages conditionnels seulement, la condition spécifiée ne sont pas validés par cette table, ils sont validés par le valideur de manœuvres. Ce dernier retourne un message d'erreur lorsque l'appareil ou le paramètre interrogé n'existent pas. Toutefois, la probabilité que l'identification de l'appareil soit erronée est faible: l'identification des appareils est obtenue du valideur de manœuvres lors de la conversion du schéma unifilaire vers une représentation 3D au démarrage du simulateur ESOPE-RV. Lorsqu'un message renferme une erreur de syntaxe, le processus **esope** envoie un message de la catégorie « erreur » au processus **audiovidéo** afin de faire entendre un

message audio indiquant le type d'erreur. Le message erroné est évidemment rejeté par le processus **esope** puis le message suivant dans la file d'attente interne est validé.

4.8.3 Conversion des messages

Avant d'être soumis au valideur de manœuvres, le message, suite à sa validation, est converti. Cette troisième étape est rendue nécessaire par la syntaxe des messages différentes utilisée par le valideur de manœuvres. En effet, ce dernier ne décode pas de messages composés de plusieurs éléments de trois caractères. Le valideur de manœuvres utilise plutôt la même syntaxe que celle employée par l'engin de validation de manœuvres lors de la validation d'une manœuvre. L'engin de validation de manœuvres peut interpréter environ 80 commandes différentes (Auxilium Technologies, 1997), qui permettent de lire un paramètre d'un appareil, de consulter l'état d'une vérification et d'obtenir les liaisons électriques d'un appareil.

La conversion des messages ne tient pas compte des deux premiers paramètres d'un message (émetteur et destinataire). La conversion débute donc avec le troisième élément du message. Cet élément associe le message à une des catégories de message possibles. La conversion étant différente selon la catégorie de message, la conversion de chaque type de message est traitée séparément dans ce document. Les catégories de message sont abordées dans l'ordre suivant: manœuvre, interrogation puis navigation.

La conversion des messages d'erreur, dont le troisième élément est « ERR » n'est pas abordée car le processus **esope** ne reçoit jamais de messages de cette catégorie.

4.8.3.1 Messages de type « manœuvre »

Ce type de message est essentiellement converti par la commande **ExecuteManœuvre**. Cette commande simule la réalisation d'une manœuvre et permet de changer la valeur d'un paramètre lorsque toutes les règles d'exploitation sont respectées. La commande **ExecuteManœuvre** est appelée avec trois arguments: le nom de l'appareil, le paramètre modifié par l'opérateur ainsi que la nouvelle valeur du paramètre. La valeur de ces arguments provient du message préalablement reçu. En effet, les éléments 4 et 6 d'un message avisant le processus **esope** d'une manœuvre représentent (voir également la section 4.7.1): la manœuvre (la nouvelle valeur du paramètre si la manœuvre est acceptée) et l'identification de l'appareil. L'identification peut être employée sans aucune conversion. L'autre élément du message reçu, soit la manœuvre, est un argument nécessaire qui doit toutefois être converti dans la syntaxe du valideur de manœuvres. Un tableau en mémoire du processus **esope** permet de convertir la manœuvre, tel qu'écrite dans le message, vers la syntaxe du valideur de manœuvres. Par exemple, l'élément de message *OPN*, qui annonce l'ouverture d'un l'appareil, signifie *Ouvert* pour le valideur de manœuvres. Le dernier argument de la commande **ExecuteManœuvre**, le nom du paramètre modifié par l'opérateur, n'est pas connu et il

ne fait pas partie du message à traduire. Heureusement, le nom du paramètre modifié peut être facilement déduit.

Pour déduire le nom du paramètre, on tire avantage du nombre limité de manœuvres qu'il est possible de faire avec le simulateur 3D. Par exemple, les seules manœuvres d'ouverture (et de fermeture) se font sur des disjoncteurs et sur des sectionneurs. Le nom du paramètre modifié est donc, dans les deux cas, le paramètre *État*. Cette même déduction peut être faite avec les autres manœuvres possibles avec le simulateur 3D. Par conséquent, il est possible d'omettre le nom du paramètre modifié du message. Tous les arguments de la commande **ExecuteManœuvre** étant connus, la commande peut être soumise à l'engin de validation du valideur de manœuvres. Par exemple, voici un message reçu par le processus **esope** devant être converti afin d'être validé par le valideur de manœuvres:

ERV ESO MAN OPN BRK 700-3

Ce message est traduit par le processus **esope** par le message suivant, pouvant être soumis au valideur de manœuvres :

ExecuteManœuvre 700-3 État Ouvert.

Les manœuvres ne sont pas tous converties par la même commande. En effet, les manœuvres consistant à vérifier l'aspect d'un appareil sont converties vers la commande **SetVerification**, plutôt que vers la commande **ExecuteManœuvre**. La commande **SetVerification** indique au valideur de manœuvres qu'une vérification de

l'aspect d'un appareil ou de l'installation a été faite. Les arguments de cette commande sont: la position de l'opérateur, le nom de l'appareil et le nom de la vérification faite. Le nom de l'appareil et le nom de la vérification sont toujours présents dans le message reçu. Cependant, le nom de la vérification nécessite une conversion avant de pouvoir être utilisée comme argument. Quant à la position de l'opérateur, celle-ci est absente du message à convertir. Toutefois, la position de l'opérateur peut être remplacée par la commande **PositionOperateur** qui permet à l'engin de validation de manœuvres d'obtenir la position courante de l'opérateur. Voici, en guise d'exemple, un message reçu par **esope** qui annonce une vérification de l'état du disjoncteur 700-3:

VBX ESO MAN VPA BRK 700-3

Ce message est converti par le processus **esope** par ce nouveau message:

SetVerification [PositionOperateur] 700-3 {Vérifié ouvert au panneau}.

La commande **PositionOperateur** retourne S ou E selon que l'opérateur est à la salle de commande ou à l'équipement.

Une dernière commande est employée au cours de la conversion d'un message indiquant une manœuvre. En effet, lorsque l'opérateur note l'heure de la dernière manœuvre par une commande vocale, le message « VBX ESO MAN TIM » est envoyé au processus **esope**. La commande équivalente, qui permet au valideur de manœuvres de noter l'heure de la dernière manœuvre, est **SetHorloge**, sans aucun argument.

4.8.3.2 Messages de type « navigation »

Le traitement des messages de navigation est plus simple que celui des messages de type manœuvre. Ces messages utilisent toujours la même commande, soit la commande **ChangementPosition**, qui permet de changer la position de l'opérateur auprès du valideur de manœuvres. Cette commande n'accepte qu'un seul argument, soit la nouvelle position de l'opérateur. La syntaxe de cet argument est: S pour la salle de commande et E pour l'équipement. Par conséquent, un changement de position de l'opérateur vers la salle de commande, dont le message reçu par le processus **esope** est « *VBX ESO NAV GTO CTR* », est toujours traduit par le nouveau message « *ChangementPosition S* ».

4.8.3.3 Messages de type « interrogation »

Il existe deux types de message d'interrogation: normal et conditionnel. Un message normal interroge un élément spécifique et le nom de l'élément fait habituellement partie du message. Le message conditionnel s'adresse plutôt à un type d'élément et a pour objectif de retourner une liste d'élément qui répond à une certaine condition. Il est ainsi possible, par l'envoi d'un message conditionnel, d'obtenir la liste des disjoncteurs qui sont fermés dans le schéma unifilaire. Ce type de message n'est pas supporté directement par l'engin de validation de manœuvres du valideur de manœuvres. En effet, il faut normalement obtenir la liste complète des éléments d'un

certain type puis filtrer cette liste pour ne garder que les appareils qui répondent à la condition initiale. Puisque le processus **esope** est en mesure d'effectuer ce traitement laborieux, les autres processus du simulateur ESOPE-RV peuvent effectuer ce genre d'interrogation sans avoir à faire ce traitement.

Les messages d'interrogation non-conditionnels emploient la commande **InfoParam** pour obtenir l'information désirée auprès du valideur de manœuvres. Les arguments de cette commande, qui proviennent tous du message reçu, sont: l'identification de l'appareil puis le paramètre à connaître. L'identification est directement à l'intérieur du message reçu et nécessite aucune conversion alors que le paramètre interrogé doit être converti dans la syntaxe du valideur de manœuvres. Par exemple, l'état d'un disjoncteur (le disjoncteur 700-1) est interrogé par l'envoi du message suivant au processus **esope**:

ERV ESO INT STE 700-1

Le message équivalent dans la syntaxe du valideur de manœuvres est:

InfoParam 700-1 État.

Les messages conditionnels sont légèrement plus compliqués à convertir car l'information demandée est obtenue en deux étapes. Dans un premier temps, la liste des appareils du type spécifié est obtenue. Cette liste est obtenue à l'aide de la commande **ListeElementsType**. Cette commande retourne tous les éléments du type spécifié par son seul argument. La deuxième étape consiste à interroger le paramètre

spécifié dans le message pour chacun des appareils retournés par la première étape. La commande **InfoParam** est employée dans cette deuxième étape afin d'interroger le paramètre spécifié dans le message. Par exemple, voici un message permettant, comme à l'exemple 4 de la section 4.7, d'obtenir les lignes dont la tension est de 735 kV:

ERV ESO INT VLT LIG VLT=735

Le message est donc converti, dans un premier temps en la commande

ListeElementsType Ligne

Puis, pour chaque nom de ligne retourné, la commande *InfoParam LX Tension* est appelée (LX prend successivement le nom des lignes, soit L1, L2, etc.).

4.8.3.4 Communs à tous les types de message

Les arguments des différentes commandes qui ne sont pas directement dans le message reçu par le processus **esope** nécessitent une traduction afin de pouvoir être compris par le valideur de manœuvres. Lorsque cette situation se présente, la traduction est faite à l'aide de la fonction `TraduitMessage` qui consulte une copie interne du fichier de ressources du valideur de manœuvres. Ce fichier contient le nom de tous les paramètres des appareils existants dans le valideur de manœuvres se retrouve à l'intérieur du processus **esope** sous la forme d'une tableau. La fonction `TraduitMessage()` permet, par exemple, de traduire la chaîne STE (obtenue d'un message) par la chaîne État (utilisée par le valideur de manœuvres) ou d'effectuer la traduction inverse. La traduction des arguments étant centralisée à l'intérieur d'une

fonction du processus **esope**, les modifications résultant du changement dans le valideur de manœuvres d'un paramètre d'un appareil sont limités à ce seul processus.

Un autre aspect commun à tout type de message est l'assignation d'un code de commande. En effet, lorsque les arguments de la commande sont décodés, le processus **esope** assigne un code numérique (1 à 100) à ce message. Ce code facilite l'interprétation du résultat obtenu en provenance du valideur de manœuvres. En effet, le format du résultat d'une commande varie considérablement d'une commande à l'autre. Certaines commandes retournent un nombre, d'autres retournent un appareil ou une liste d'appareil. Pour cette raison, chaque commande pouvant être envoyée au valideur de manœuvres est associée à un code numérique. À la réception du résultat d'une commande, ce code permet de déterminer le format du résultat obtenu.

4.8.4 Envoi de la commande au valideur

La quatrième étape, suite la traduction du message, est l'envoi de la commande à ce au valideur de manœuvres. Le message contenant la commande est donc envoyé au valideur de manœuvres. Le temps requis par ce dernier pour lire un message, effectuer son traitement puis retourner le résultat varie de quelques millisecondes à près de 10 secondes sur l'ordinateur normalement utilisé. Pendant le traitement de la commande par le valideur de manœuvres, le processus **esope** effectue seulement la lecture des messages provenant des processus autres que le valideur de manœuvres. Les messages

lus sont ajoutés à la file d'attente interne jusqu'à ce que le valideur de manœuvres renvoie le résultat du message qu'il traite.

4.8.5 Lecture des résultats provenant du valideur

Le résultat de la dernière commande est retourné au processus **esope** par le valideur de manœuvres après quelques instants. Le message contenant le résultat ayant été détecté par la fonction `select()`, chacun des liens vers les autres processus sont vérifiés de nouveau afin de trouver le lien sur lequel un message est en attente. La vérification des liens débutant toujours par le lien entre le valideur de manœuvres et le processus **esope**, le message en attente à cet endroit est immédiatement détecté puis lu à l'aide de la commande `read()`.

4.8.6 Validation du résultat

À la sixième étape, le message lu provenant du valideur de manœuvres est traité par **esope** afin de s'assurer qu'il est valide car il arrive parfois que les messages reçus ne contiennent qu'un ou deux retours de chariot (code ASCII 10). Ces messages invalides sont causés par la façon différente que possèdent UNIX et Windows de traiter les chaînes de caractères: Windows ajoute toujours un retour de chariot et un changement de ligne à la fin d'une chaîne tandis qu'UNIX se contente d'ajouter un retour de

chariot seulement. Par conséquent, lorsqu'un processus sur UNIX (comme le processus **esope**) lit un message provenant du valideur, la chaîne est lue jusqu'au premier retour de chariot. Cependant, le second caractère, le changement de ligne, n'est pas lu immédiatement. Le prochain appel à la fonction `select`, permettant de déterminer les liens avec un message en attente, indiquera qu'un message est encore en attente. Ce message n'est en fait que le caractère de changement de ligne. Le message erroné est lu, sans toutefois être traité par le processus **esope**. Lorsque le message lu est valide, la fonction `AnalyseReponse` est appelée afin de faire l'analyse de la réponse du valideur de manœuvres. Le premier test effectué par cette fonction est de s'assurer qu'une erreur ne soit pas survenue dans l'engin de validation de manœuvres du valideur de manœuvres.

L'engin de validation de manœuvres peut retourner deux types d'erreur possibles: les erreurs de commandes et les erreurs de manœuvres. Le premier type d'erreur, les erreurs de commandes, se produisent lorsqu'une commande de l'engin de validation de manœuvres, `InfoParam` par exemple, est utilisée en ne respectant pas la syntaxe établie (position des arguments, nombres d'arguments). Ce type d'erreur est également produit lorsqu'un appareil inexistant ou une vérification inexistante est utilisé comme argument d'une commande de l'engin de validation de manœuvres. Les erreurs de manœuvres, le second type d'erreur, font suite au non-respect par l'opérateur d'une règle d'exploitation du valideur de manœuvres.

Lorsqu'un de ces deux types d'erreur se produit, le processus **esope** reçoit un message contenant un message d'erreur plutôt que le résultat de la dernière commande. Le tableau 4-5 présente les différents messages d'erreur possibles ainsi que leur signification.

Tableau 4-5 Messages d'erreur du valideur de manœuvres

Message d'erreur reçu	Description	Code d'erreur interne
invalid command name::" XYZ "	La commande XYZ n'existe pas	10001
L'élément identifié <XYZ> n'est pas reconnu.	L'élément XYZ n'existe pas	10002
Paramètre inexistant...	Un argument de la commande est absent	10003
Nombre de paramètres invalide pour la fonction <XYZ>.	Le nombre d'argument est invalide pour la commande XYZ	10005
Paramètre <XYZ> invalide	L'argument XYZ est invalide	10004
-1	La commande est inutile	10008
0	Une erreur de manœuvre (non-respect d'une règle) a été commise	Pas de code d'erreur.

Les six premiers messages de ce tableau sont des erreurs de commande. Le septième message indique qu'une erreur de manœuvre s'est produite. Comme l'indique le tableau 4-5, le message d'erreur reçu par **esope** en provenance du valideur de manœuvres est une phrase explicative de l'erreur. Les erreurs sont toutes associées à un code d'erreur interne utilisé dans le but de faire entendre un message audio expliquant l'erreur à l'opérateur. Cet aspect est présenté plus loin dans cette section. Les erreurs du second type, les erreurs de manœuvres, exigent une étape de plus afin

de connaître l'erreur exacte commise par l'opérateur. En effet, lorsqu'une manœuvre est soumise au valideur avec la commande **ExecuteManœuvre**, cette dernière retourne uniquement 1 ou 0. Un 1 signifie que la manœuvre a été acceptée tandis qu'un 0 signifie que la manœuvre n'est pas acceptée car elle ne respecte pas une règle (sans dire laquelle).

Pour connaître la règle non-respectée par l'opérateur, il faut envoyer au valideur de manœuvres un autre message avec la commande **set NomRegleExec**. Cette commande retourne le contenu de la variable **NomRegleExec** de l'engin de validation de manœuvres qui contient le nom de la dernière règle échouée par l'opérateur. La figure 4-3 présente le traitement nécessaire afin d'obtenir le type d'erreur de manœuvre.

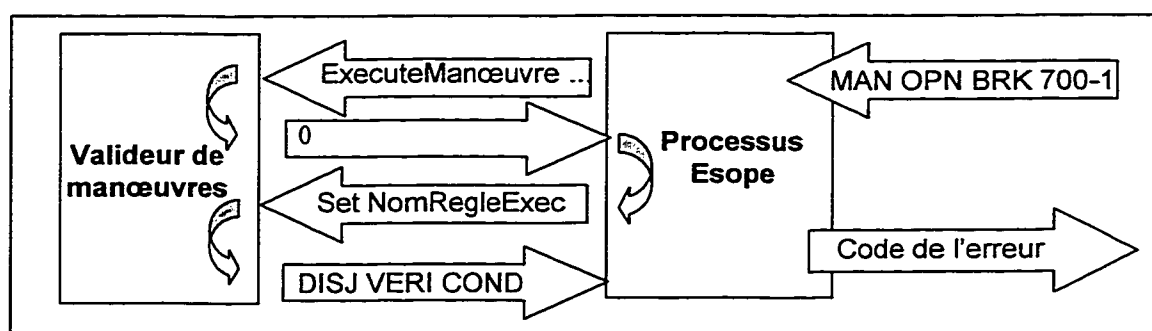


Figure 4-3 Traitement des erreurs de manœuvre

Le nom de la règle permet par la suite de déduire l'erreur commise par l'opérateur. En effet, le nom de chaque règle renseigne sur l'aspect qui est validé par cette règle. Par exemple, la règle **DISJ VERI COND** (comme à la figure 4-3) s'assure que les conditions d'exploitation (**COND**) du poste ont été vérifiées (**VERI**) avant de

manœuvrer un disjoncteur (DISJ) alors que la règle SECT VERI COND valide également les conditions d'exploitation mais pour les manœuvres sur les sectionneurs (SECT). Pour les deux exemples, l'erreur de l'opérateur est de n'avoir pas vérifié les conditions d'exploitation avant d'effectuer sa manœuvre. Par conséquent, afin de réduire le nombre de cas d'erreurs possibles, ces deux règles sont traitées comme si elles étaient une même erreur. Le tableau 4-6 présente les différentes erreurs possibles en fonction du nom de la règle échouée.

Tableau 4-6 Erreurs de manœuvres possibles

Nom de la règle échouée	Description de l'erreur	Code d'erreur interne
* MTTE	Il faut mettre le mode local de l'appareil à Télé	10010
* HORL *	Il faut noter l'heure de la dernière manœuvre avant de poursuivre	1501
SECT VERI A FAIR *	Il faut vérifier l'état du dernier sectionneur manoeuvré avant de poursuivre	1004
DISJ VERI A FAIR *	Il faut vérifier l'état du dernier disjoncteur manoeuvré avant de poursuivre	1002
* VERI COND	Il faut vérifier les conditions d'exploitation du poste avant de poursuivre	1001
* INSP ZON *	Il faut inspecter la zone concernée du poste avant de poursuivre	201
* MTLO	Il faut mettre le mode local de l'appareil à Local	10007

Une étoile dans les noms de règle signifie que le nom de la règle contient en fait un élément de plus (SECT, DISJ, etc.) mais que l'erreur est la même, peu importe la valeur de cet élément. Le nom de la règle échouée est obtenu en ajoutant un message

bidon au début de la file d'attente de messages du processus **esope** qui provoque l'envoi de la commande **set NomRegleExec** au valideur de manœuvres. Lorsque ce dernier retourne le résultat de cette commande, le traitement de l'erreur peut se poursuivre car la nature de l'erreur de l'opérateur est maintenant connue.

Lorsqu'une erreur est détectée, un code d'erreur interne, différent pour chacune des erreurs des tableaux 4-4 et 4-5, est assigné au message. Par exemple, le message indiquant que le mode de commande de l'appareil doit être à Local avant de faire la dernière manœuvre est associé au code d'erreur 10007. Ce code numérique est également associé à un message audio qui décrit l'erreur de manœuvre effectuée par l'opérateur.

4.8.7 Conversion et envoi du résultat

Les cas d'erreurs étant écartés, le message reçu du valideur de manœuvres contient donc le résultat de la dernière commande lui ayant été soumise. Ce résultat peut maintenant être converti afin d'être retourné au processus émetteur. Tel que spécifié à la section 4.7, le format du message contenant le résultat est très semblable au message d'origine. Pour la plupart des cas, deux éléments changent de position dans le message. De plus, pour les messages d'interrogation, le résultat est ajouté à la fin du message à retourner au processus émetteur. Le tableau 4-7 présente les différents résultats possibles en fonction de l'interrogation.

Tableau 4-7 Résultats possible d'une interrogation

Interrogation normale	Résultat	Interrogation conditionnelle	Résultat
CUR	Un nombre	CUR=XX	une liste d'appareils
DPU	<i>Entrée ou Sortie</i>	DPU =XX	une liste d'appareils
MLO	<i>Direct ou Distance</i>	MLO =XX	une liste d'appareils
MTE	<i>Télé ou Local</i>	MTE =XX	une liste d'appareils
NBR	Un nombre	NBR =XX	une liste d'appareils
PAC	Un nombre	PAC =XX	une liste d'appareils
PRE	Un nombre	PRE =XX	une liste d'appareils
STE	<i>Ouvert ou Fermé</i>	STE =XX	une liste d'appareils
VL1	Un nombre	VL1=XX	une liste d'appareils
VL2	Un nombre	VL2=XX	une liste d'appareils
VL3	Un nombre	VL3=XX	une liste d'appareils
VLT	Un nombre	VLT =XX	une liste d'appareils
XST	<i>Oui ou Non</i>	XST =XX	une liste d'appareils

Cependant, les messages d'interrogation conditionnels sont traités différemment. Ces messages exigent l'envoi de plusieurs commandes successives au valideur de manœuvres afin de graduellement construire la liste d'appareils à retourner (voir la figure 4-4)

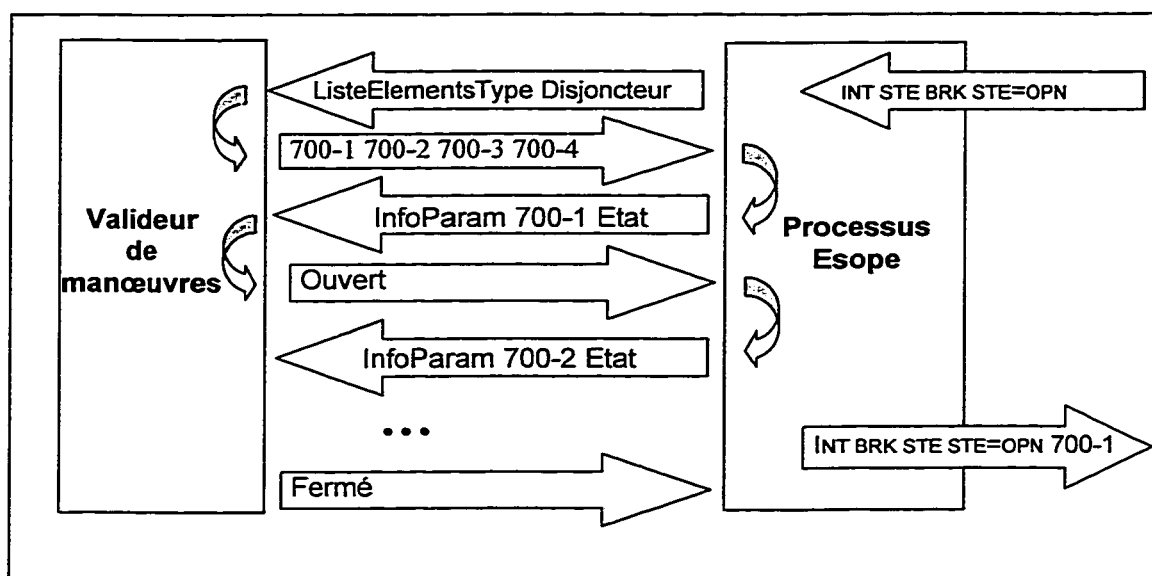


Figure 4-4 Messages conditionnels

Dans un premier temps, on obtient la liste des appareils d'un certain type. Ensuite, pour chaque appareil retourné, on compare ses paramètres avec la condition spécifiée par le processus émetteur. Avant de retourner la liste d'appareil au processus émetteur, il faut s'assurer que tous les éléments du type spécifié ont été interrogés. S'il reste d'autres appareils, on poursuit l'interrogation des appareils.

Pour les messages contenant une erreur de commande ou une erreur de manœuvre, le traitement diffère quelque peu. En effet, pour faire entendre le message audio associé à l'erreur, il faut envoyer le code de l'erreur au processus ayant la responsabilité de faire entendre les messages audio, soit le processus **audiovidéo**. Le processus à l'origine de ce message n'a pas à être avisé de l'erreur. En effet, les manœuvres effectuées à partir

de l'interface de reconnaissance vocale passe par le valideur de manœuvres pour ensuite être acheminées, si elles sont acceptées, au processus en charge de la représentation 3D, soit **esope-rv**, afin d'être intégrées au poste virtuel. Les manœuvres effectuées directement par la représentation 3D (à l'aide de la main virtuelle, par exemple) doivent également être acceptées par le valideur avant de retourner au processus **esope-rv** et d'être intégrées au poste virtuel (par le processus **worldtoolkit**).

Suite à l'envoi du résultat au processus émetteur, le message est effacé de la file d'attente de messages du processus **esope**.

4.9 Conclusion

La création du processus **esope**, agissant comme interface entre le valideur de manœuvres et les autres processus offre de nombreux avantages. Par exemple, il permet de créer de nouvelles commandes disponible aux processus d'ESOPE-RV en combinant des commandes existantes du valideur de manœuvres. Le traitement requis par ces combinaisons de commandes est effectué par le processus **esope**, ce qui permet aux autres processus de se concentrer uniquement sur leur tâche respective et d'accélérer le déroulement de la simulation 3D. Le prochain chapitre analyse principalement de façon qualitative les performances du simulateur ESOPE-RV lors de la réalisation du plan de manœuvres type.

5. Performances d'ESOPE-RV

5.1 *Introduction*

Le projet ESOPE-RV se donne comme objectif général la réalisation d'un simulateur de manœuvres en réalité virtuelle. Afin de démontrer la faisabilité de ce projet en tant qu'outil de formation, il est essentiel que le niveau de performance globale du simulateur permette de réaliser une simulation réaliste. Par conséquent, il est nécessaire que les différents critères d'évaluation d'une simulation en réalité virtuelle ainsi que la rapidité d'exécution de l'ensemble des processus soient assez élevés au moment de la réalisation d'un plan de manœuvre. Le chapitre présent traite donc du niveau de performance globale d'ESOPE-RV lors du démarrage, de la conversion du schéma unifilaire et de la réalisation d'un plan de manœuvres. Les résultats présentés démontrent que, malgré les nombreuses contraintes reliées au projet, ce dernier atteint un niveau de performance acceptable.

5.2 *Démarrage du simulateur*

Le démarrage est le moment où le script de démarrage est lu, permettant ainsi le lancement des processus impliqués dans le simulateur ESOPE-RV. Suite au démarrage d'un processus, le script effectue habituellement une pause afin de permettre au processus nouvellement exécuté de se déplacer en mémoire et de débiter l'initiation

des différents liens de communications. La pause est habituellement de cinq secondes mais certains processus ont une pause d'une durée différente: 10 secondes pour le processus **esope-rv** alors qu'il n'y a pas de pause pour le processus **esope**. La durée de cette pause dépend de plusieurs facteurs: la puissance des stations de travail, la vitesse de transmission permise sur le réseau informatique Ethernet et le nombre de liens à initialiser. Avec l'équipement informatique utilisé par le simulateur ESOPE-RV, une pause totale de 40 secondes semble donner un excellent rendement. En ajoutant le temps des différentes pauses au temps de lecture et d'exécution du script, le temps total nécessaire pour lancer les différents processus est d'environ 45 secondes. Ce temps ne tient toutefois pas compte du démarrage du valideur de manœuvres, qui n'est pas lancé automatiquement par le script de démarrage mais plutôt manuellement par l'utilisateur, avant même le lancement du script de démarrage. Le temps requis pour lancer le valideur et pour charger le schéma type en mémoire (voir l'annexe 2) est d'environ 5 secondes. Le temps total pour démarrer tous les processus est donc de 50 (45 + 5) secondes environ.

5.3 *Création de l'installation en réalité virtuelle*

Au moment où les processus sont lancés et que les différents liens de communication sont établis, la conversion du schéma unifilaire peut débuter. Tel qu'expliqué précédemment (voir la section 2.4), cette conversion, effectuée par le processus **worldtoolkit**, permet de convertir le schéma unifilaire chargé dans le valideur de

manœuvres en une installation électrique équivalente en réalité virtuelle. Le schéma unifilaire utilisé pour la conversion est un schéma type, présenté en annexe. Ce schéma représente un poste d'interconnexion de haute tension (735 kV) et il contient deux lignes d'entrées et deux lignes de sorties à cette même tension. Une partie de la puissance est détournée vers deux lignes, dont la tension est de 120 kV. La tension est abaissée de 735 kV à 120 kV par deux transformateurs. Ce schéma unifilaire contient également 10 disjoncteurs et 20 sectionneurs.

La conversion de ce schéma en un poste équivalent en 3D se fait en plusieurs étapes. Toutefois, deux de ces étapes utilisent à elle seules presque tout le temps requis par la conversion. Ces deux étapes sont: l'obtention par le processus **worldtoolkit** des données du schéma unifilaire auprès du valideur de manœuvres et le chargement dans l'environnement virtuel des différents objets 3D représentant les appareils du poste, également effectué par le processus **worldtoolkit**. Au cours de la première étape, le valideur de manœuvres est consulté afin d'obtenir de l'information sur les appareils du schéma unifilaire à convertir. Cette étape est la plus intense quant à la quantité de messages échangés entre les différents processus. En effet, environ 150 messages sont à ce moment échangés entre le valideur de manœuvres et le processus **worldtoolkit**. Évidemment, les messages passent aussi par les deux processus servant d'interface au valideur de manœuvres et à **worldtoolkit**: les processus **esope** et **ESOPE-RV**. La durée de ces échanges est d'environ 25 secondes.

La conversion du schéma unifilaire est terminée lorsque chaque objet 3D de l'installation (transformateurs, disjoncteurs, etc.), emmagasiné sur le disque rigide, est chargé en mémoire et positionné dans l'environnement virtuel. Un aspect intéressant de l'outil de création de l'environnement virtuel, Worldtoolkit, est sa capacité de sauvegarder sur disque des objets 3D créés avec une combinaison de polygones. Il est beaucoup plus rapide, lors de la création du poste virtuel, de charger les objets 3D à partir du disque rigide que de créer dynamiquement chacun des polygones et de les positionner de façon à recréer l'objet 3D. Le chargement des objets 3D ne se fait pas de façon instantanée car il implique de nombreux accès au disque rigide, réputé pour être un support de données lent, comparativement à la mémoire vive d'un ordinateur¹. Il faut patienter environ 10 secondes avant que tous les objets 3D soient chargés et positionnés dans le poste virtuel. La conversion du schéma par le processus **worldtoolkit** exige donc 35 (25 + 10) secondes au total, du début des échanges avec le valideur de manœuvres jusqu'à l'affichage à l'écran du poste virtuel en 3D.

5.4 Réalisation du plan de manœuvres type

La réalisation du plan de manœuvres type présenté à l'annexe 1 permet d'étudier les performances du simulateur ESOPE-RV en tant qu'outil de validation de manœuvres. Contrairement aux deux autres items analysés dans ce chapitre, soit le démarrage et la

¹ Le temps d'accès à la mémoire vive est de 60-70 ns alors qu'il est de 10 ms (ou 10 000 000 ns) pour un disque rigide. Le taux de transfert des données est aussi plus élevé lors de la lecture en mémoire vive.

conversion du schéma, où le temps d'exécution requis doit être le plus court possible, la réalisation d'un plan de manœuvres ne peut être analysée de la même façon. En effet, le simulateur ESOPE-RV cherche à reproduire de façon réaliste l'exécution d'un plan de manœuvres dans un poste. Si le simulateur reproduisait exactement cette réalité, le temps nécessaire serait rigoureusement le même. Toutefois, une reproduction exacte dans le temps n'est pas nécessairement souhaitable ou utile. Imaginons par exemple, un déplacement à l'extérieur, dans la cour aux équipements du poste, qui requiert normalement une marche de deux minutes à un opérateur dans un poste réel. Il y a peu d'intérêt à ce que ce même déplacement dans l'environnement virtuel exige également deux minutes. En effet, l'opérateur risque de trouver ses déplacements inutilement long et de perdre de son intérêt pour le simulateur ESOPE-RV.

Les déplacements de l'opérateur dans le poste virtuel sont ainsi plus rapides que dans la réalité. En effet, dans l'environnement virtuel, l'opérateur se déplace toujours à un pas de course, ce qui lui permet quand même de prendre conscience de la grande étendue du poste virtuel. Alors que les déplacements sont plus rapides, les manœuvres sont quant à elles plus lentes à réaliser dans l'environnement virtuel. Cette lenteur est causée par l'utilisation du valideur de manœuvres, qui exige de nombreux accès aux données sur le disque rigide de l'ordinateur de type PC. Ces accès ralentissent parfois considérablement la validation des manœuvres de l'opérateur. Ce ralentissement varie d'une manœuvre à une autre. Par exemple, avec un ordinateur ayant un microprocesseur de type Intel 80486 cadencé à 33 Mhz, l'ouverture d'un disjoncteur

est validée en 10 secondes alors que la vérification de l'état d'un appareil (disjoncteur ou sectionneur) en requiert moins d'une. Cependant, avec un ordinateur de type PC plus rapide, les délais pour valider la manœuvre seraient beaucoup plus courts. Par exemple, l'ouverture d'un disjoncteur est validée en seulement 1 seconde sur un ordinateur IBM de type PC équipé d'un microprocesseur Pentium 133 Mhz. L'utilisation d'un ordinateur plus performant permettrait d'accélérer de beaucoup la validation des manœuvres par le valideur de manœuvres d'OPÉRATEUR+.

Le temps d'exécution des manœuvres et des déplacements n'est pas le seul critère d'évaluation de la performance du simulateur ESOPE-RV lors de la réalisation d'un plan de manœuvres. Le taux de rafraîchissement ainsi que le temps de latence sont également pertinents dans l'évaluation d'un environnement en réalité virtuelle. Le taux de rafraîchissement est une mesure du nombre d'images différentes qu'il est possible de générer en une seconde. À la télévision et au cinéma, les taux de rafraîchissement sont de 30 et 24 images par seconde respectivement. Avec un taux d'au moins 20 images par seconde, le cerveau combine ces images et donne l'impression d'obtenir un mouvement fluide (Isdale, 1993). Plus le nombre d'images est élevé, plus les déplacements dans la simulation en réalité virtuelle sont fluides et réalistes. Toutefois, lorsque le taux de rafraîchissement est faible, les déplacements se font de façon saccadés et le réalisme de l'environnement virtuel est moindre.

Pour ce qui est du temps de latence, ce dernier est le temps requis par l'ordinateur pour traiter un déplacement et rafraîchir l'environnement virtuel. Idéalement, ce temps serait de zéro seconde: les déplacements sont dans ce cas immédiatement visibles dans l'environnement virtuel. Lorsque le temps de latence est trop élevé, l'utilisateur peut avoir de la difficulté à se déplacer dans l'environnement virtuel. Lorsque cette situation se produit, les déplacements de l'utilisateur ne sont pas immédiatement intégrés à l'environnement virtuel. L'utilisateur a alors tendance à répéter la commande qui lui permet de se déplacer. Quand les déplacements sont finalement intégrés à l'environnement virtuel, l'utilisateur s'aperçoit qu'il s'est déplacé trop loin. Afin de corriger sa position, il décide de faire le déplacement inverse. Mais étant donné que son déplacement n'est pas immédiatement intégré à l'environnement virtuel, il répète également son déplacement inverse, pour finalement s'apercevoir encore qu'il n'est pas exactement à l'endroit où il voulait être. Ce problème est surtout apparent lors de déplacements ou de mouvement fins dans le but d'atteindre une position précise. Un temps de latence trop élevé cause parfois des maux de tête, des nausées ou des étourdissements, surtout lorsque qu'un casque à écrans intégrés est utilisé. Selon Thomas Furness de l'Université de Washington à Seattle, un temps de latence de 16 ms est idéal lorsque ce type de casque est employé (Johnson, 1995). Parce que le simulateur ESOPE-RV n'utilise pas ce type de casque et que dans l'ensemble, son degré d'immersion est assez faible, un temps de latence plus élevé (250 ms) est acceptable sans que des effets physiologiques néfastes ne soient causés.

Les performances de la simulation lors de la réalisation de manœuvres en réalité virtuelle varie beaucoup en fonction de la quantité d'objets 3D présentés à l'utilisateur. Lorsque le nombre d'objets est bas (moins de 20 appareils, l'utilisateur est à l'extérieur du poste), le taux de rafraîchissement est autour de 14 images à la seconde. Cette bonne performance est possible grâce à l'utilisation de représentations 3D simplifiées des appareils du poste. Les différents niveaux de détails pour la représentation des appareils, où les appareils éloignés de l'utilisateur ne sont pas représentés ou encore ils le sont uniquement par un cube, contribuent également à obtenir ce taux de rafraîchissement élevé. Le taux de rafraîchissement lorsque l'opérateur est à la salle de commande est également élevé et il varie entre 10 et 14 images par secondes selon le panneau présenté à l'écran. Toutefois, les performances sont plus faibles lorsqu'un nombre élevé d'appareils sont présentés à l'écran (plus de 40 appareils, l'utilisateur est à l'extérieur du poste). Au cours de cette situation, le taux de rafraîchissement est d'environ 5 images par secondes. Le taux peut descendre jusqu'à 2 images par secondes lorsque l'utilisateur est très près d'un nombre élevé d'appareils (plus de 40 appareils, l'utilisateur est à l'extérieur du poste). Ce faible taux de rafraîchissement est obtenu malgré des efforts de limiter un minimum le nombre d'appareils dans le poste virtuel ainsi que leur niveau de complexité. La faible puissance du poste de travail SGI R4000 pour le traitement d'objets 3D explique ce niveau de performance.

Le temps de latence du simulateur ESOPE-RV est très bas, soit de moins de 250 millisecondes. De plus, la quantité d'appareils visibles à l'écran n'influence pas ce temps de latence.

5.5 Conclusion

Les performances du simulateur ESOPE-RV sont en général acceptables. Le temps de latence est très bas alors que le taux de rafraîchissement est généralement à plus de 10 images par secondes. Cependant, les performances sont parfois décevantes lorsqu'un nombre élevé d'appareils sont visibles à l'écran. Malgré les efforts faits pour minimiser le traitement nécessaire, la capacité de traitement maximale du poste de travail est à ce moment atteinte. Cette faible performance ne gêne pas la réalisation d'un plan de manœuvres mais affecte un peu la sensation d'immersion que laisse transparaître la simulation. Le prochain chapitre aborde de nouveau les performances du simulateur ESOPE-RV, ainsi que le projet dans son ensemble. Des modifications sont également proposées afin d'augmenter la sensation d'immersion dans le poste virtuel.

6. Limitations

6.1 *Introduction*

La simulation en réalité virtuelle requiert des ordinateurs et des stations de travail très puissants. Avec de tels ordinateurs, l'environnement virtuel peut être d'un réalisme saisissant car les objets 3D y sont particulièrement détaillés par le biais de polygones auxquelles s'ajoutent différentes textures. Cependant, les ordinateurs employés pour réaliser le simulateur ESOPE-RV étant un peu dépassés, la représentation des appareils du poste virtuel a dû être simplifiée afin d'obtenir des performances acceptables. Ce chapitre présente donc les nombreuses limitations imposées à ce projet par le matériel informatique. De plus, les limitations causées par les périphériques ainsi que l'outil de création de l'environnement virtuel y sont précisés. Les contraintes fonctionnelles du simulateur 3D sont par la suite présentées.

6.2 *limitations de type matériel*

L'équipement employé pour la réalisation de ce projet a été prêté par Hydro-Québec. Cet équipement se compose de trois ordinateurs de marque Silicon Graphics. Ces ordinateurs sont âgés de cinq (5) à sept (7) ans et ne possèdent pas la puissance nécessaire pour l'application de textures (texture mapping) sur les objets 3D de l'environnement virtuel ou pour la gestion d'un nombre élevé de ce type d'objet. Par

conséquent, la simulation du poste en réalité virtuelle est réalisée sans l'utilisation de textures ainsi qu'avec un minimum de détail: les appareils moins utilisés ou ceux que l'opérateur n'a pas à utiliser ne sont pas représentés dans le poste et les appareils plus communs, tels les transformateurs, les sectionneurs et les disjoncteurs, sont représentés de façon simplifiée dans l'environnement virtuel. De plus, à l'exception de ceux près des transformateurs haute tension, les fils aériens ne sont pas représentés.

La simulation en réalité virtuelle est souvent utilisée conjointement avec des lunettes ou un casque. En effet, ces périphériques, qui projettent l'image de la simulation devant les yeux, permettent d'augmenter la sensation d'immersion dans l'environnement virtuel. Ce genre d'équipement, dont le coût varie de 1000 à 10000 dollars (été 1995), était trop onéreux pour le projet lors de sa réalisation en 1994-1995. La simulation de manœuvres est donc plutôt du type *fish tank* et se fait sur un moniteur d'ordinateur de grande dimension. Des lunettes polarisantes, qui ont l'apparence d'une paire de lunettes de soleil, servent à donner une impression de profondeur à la simulation.

Le système de reconnaissance vocale est également sujet à certaines limitations. En effet, le système est de type « dépendant de l'utilisateur » et exige que chaque utilisateur entraîne le système de reconnaissance vocale à reconnaître son timbre de voix. Heureusement, cet entraînement, d'une durée moyenne de 90 minutes, n'est réalisé qu'une seule fois.

6.3 Limitations de type logiciel

Un logiciel commercial est utilisé pour assister à la gestion de la simulation 3D. Le temps économisé par l'emploi de ce produit, WorldToolKit de Sense8, a permis de développer d'autres aspects de la simulation. En contrepartie, cette avenue offre moins de flexibilité et aucune possibilité d'adaptation aux besoins particulier de ce projet de recherche. Par exemple, le gant PowerGlove de Mattel n'est pas directement supporté par WorldToolKit. Afin de l'intégrer au projet ESOPE-RV, un gestionnaire a été créé pour gérer directement le port de communication de l'ordinateur sur lequel le gant est branché.

La conversion de schémas unifilaires est également soumis à certaines restrictions. L'information sur le poste et ses appareils est obtenue d'un schéma unifilaire chargé dans le module de validation de manœuvres du logiciel OPÉRATEUR+. Pour être en mesure d'effectuer une simulation dans un poste particulier, il est essentiel que l'information sur ce poste soit dans un fichier lisible par les logiciels de la gamme OPÉRATEUR+. L'éditeur de schéma d'OPÉRATEUR+ permet de générer ce type de fichier. Toutefois, la génération automatique d'un poste en 3D est possible uniquement si la configuration de barres du schéma unifilaire est de type « *double bus - double breaker* ». Les autres configurations, telle « *Ring bus* », etc., ne sont pas supportées pour l'instant. Suite au processus de conversion, seuls les appareils les plus importants sont représentés dans le poste 3D. Les disjoncteurs, les sectionneurs, les

transformateurs et les pylônes de même qu'une partie du câblage y sont donc visibles. Les autres appareils tels les transformateurs de courant et de tension, les parafoudres, les turbines, les compensateurs ne sont pas supportés.

Contrairement à la cour du poste, la salle de commande du simulateur de postes n'est pas générée de façon automatique à partir du schéma unifilaire. Le temps requis pour réaliser un programme de conversion permettant de générer la salle de commande à partir du schéma unifilaire étant trop élevé (environ 4 mois-personne), les efforts ont plutôt été concentrés sur la construction automatique de la cour du poste. La salle de commande a plutôt été réalisée à l'aide d'un logiciel de dessin 3DSTUDIO de Alias/WaveFront.

La variété de manœuvres qu'il est possible de réaliser avec ESOPE-RV est moindre qu'avec le valideur de manœuvres d'OPÉRATEUR+. Le manque de temps a fait en sorte qu'une quantité limitée de manœuvres ont été intégrées au simulateur ESOPE-RV. Les manœuvres que l'opérateur peut présentement effectuer sur les appareils d'ESOPE-RV consistent à modifier leur mode de commande et leur état ou encore à réaliser les quatre vérifications les plus communes dans un plan de manœuvres chez Hydro-Québec. Ces vérifications consiste à vérifier les conditions d'exploitation, vérifier l'état d'un appareil au sémaphore, vérifier l'état d'un appareil au panneau synoptique et inspecter la zone concernée par la manœuvre.

Lorsque l'engin de validation de manœuvres détecte une fausse manœuvre de la part de l'opérateur en formation, un message auditif décrivant l'erreur est aussitôt entendu. Alors que l'engin de validation peut détecter quelques centaines d'erreurs de manœuvre, seulement une vingtaines de messages auditifs ont été enregistré. Toutefois, un message audio est disponible pour toutes les erreurs de manœuvres qu'il est possible de faire avec le simulateur 3D.

Malgré ces limitations, le simulateur de postes ESOPE-RV permet tout de même de réaliser un plan de manœuvres réaliste. Pour l'opérateur en formation, il devient possible de marcher dans la cour du poste d'une extrémité à l'autre afin d'évaluer l'ampleur de celui-ci ainsi que le nombre d'appareils. Par exemple, l'opérateur a la possibilité de se tenir à côté d'un transformateur, de lire l'étiquette pour l'identifier et, par la dimension et la forme du transformateur, évaluer le niveau de tension à cet endroit. Il peut également identifier les barres à proximité qui sont normalement sous tension. Après quelques minutes dans le poste, l'opérateur est donc conscient des appareils devant être manoeuvrés pour isoler une pièce d'équipement. La salle de commande du poste étant aussi représentée, l'opérateur a la possibilité d'entrer dans cette salle afin d'observer les panneaux synoptiques où sont situés les mécanismes permettant de modifier l'état des appareils. Dans cette salle, l'opérateur peut exécuter les opérations suivantes: manipuler un levier ou un bouton pour changer l'état d'un appareil ou son mode de commande, lire les valeurs électriques ainsi que l'état des principaux équipements. Par une commande vocale, il peut effectuer les vérifications

exigées par les règles d'exploitation d'Hydro-Québec. L'ensemble de ces actions lui permet éventuellement de compléter un plan de manœuvres dans le poste virtuel.

6.4 Conclusion

Les limitations imposées au projet ESOPE-RV proviennent surtout de l'équipement informatique, plus précisément de la faible puissance des ordinateurs disponibles lors de la réalisation du projet. Avec un équipement informatique plus puissant, le réalisme des appareils de même que le nombre d'appareils supporté seraient immédiatement augmentés. Cependant, ces limitations n'empêchent pas la réalisation d'un plan de manœuvres avec le simulateur ESOPE-RV. En effet, les éléments nécessaires à la réalisation du plan de manœuvres type sont tous présents. Il est possible de réaliser des manœuvres, de se déplacer dans le poste et surtout d'apprendre grâce aux messages d'erreurs audio et au support multimédia.

7. Conclusion

Le projet ESOPE-RV vise la réalisation d'un logiciel de formation des opérateurs utilisant la réalité virtuelle. Ce logiciel améliore la formation de l'opérateur de plusieurs façons en lui permettant d'effectuer des manœuvres dans un environnement virtuel qui reproduit la réalité d'un poste. Les manœuvres réalisées dans cet environnement sont constamment validées par l'engin de validation de manœuvres d'OPÉRATEUR+. Lorsque l'opérateur commet une erreur de manœuvre, un message audio lui indique la raison de son erreur. L'opérateur dispose d'une aide multimédia composée de photos d'appareils présents dans un poste. De plus, des séquences vidéo affichant les mêmes appareils pendant l'exécution d'une manœuvre sont également disponibles. Le logiciel ESOPE-RV accepte un grand nombre de postes différents, ce qui en fait un outil de formation très flexible. En effet, en permettant à l'opérateur de réaliser des manœuvres dans des postes différents, la variété des manœuvres possibles est grandement augmentée.

Plusieurs techniques ou technologies ont été mises à contribution dans la concrétisation du projet ESOPE-RV afin d'en faire un logiciel complet, flexible et inédit. Voici un résumé de ces techniques ou technologies:

- la conversion automatique d'un schéma unifilaire en une représentation 3D: à partir d'un schéma unifilaire d'un poste chargé dans le valideur de manœuvres, une représentation réaliste en 3D de ce poste est automatiquement construite dans

l'environnement virtuel, en appliquant les mêmes critères utilisés pour la construction de postes réels. Cette conversion évite de créer manuellement chaque poste dans l'environnement virtuel;

- La réalité virtuelle: utilisée pour reproduire un poste de façon réaliste à l'aide d'un ordinateur. La formation de l'opérateur se fait ensuite à l'aide de cette représentation;
- La commande vocale: la commande vocale permet de transmettre une commande au simulateur 3D en utilisant la voix. La commande vocale libère les mains de l'utilisateur, qui peuvent être utilisées à d'autres fins tel la manipulation de la main virtuelle par le gant;
- Le gant: employé pour faciliter l'interaction avec l'environnement virtuel en permettant à l'opérateur d'utiliser sa main afin de manier la main virtuelle de la salle de commande. Cette main est utilisée pour manipuler les leviers et les boutons de la salle de commande afin de réaliser une manœuvre sur une appareil;
- L'immersion sonore: utilisé afin d'augmenter le réalisme de l'environnement virtuel par l'emploi de bruits qu'il est habituellement possible d'entendre dans un poste réel. Par exemple, les sons servent d'indice sonore permettant de confirmer l'exécution d'une manœuvre;
- L'aide multimédia: l'opérateur dispose d'une banque de photos et de vidéos d'appareils présents dans un poste. Ce support multimédia permet à l'opérateur de faire le lien entre la représentation des appareils dans l'environnement et les appareils réels;

- L'intégration du valideur de manœuvres d'OPÉRATEUR: la validation des manœuvres de l'opérateur ainsi que les calculs de courants, de tension et de puissance sont effectués par le valideur de manœuvres d'OPÉRATEUR+. Cette intégration permet de rapidement avoir accès à un algorithme d'écoulement de puissance mature et rapide et d'avoir accès également à un mécanisme de validation de manœuvres flexible. Cette intégration est également l'objet principal de ce mémoire.

Le développement d'ESOPE-RV a été fait en tentant toujours de maximiser l'utilisation des postes de travail. En effet, la conception modulaire du projet permet de répartir la puissance requise sur les ordinateurs disponibles. Le module le plus exigeant, soit celui responsable de la représentation 3D, peut alors être exécuté seul sur le poste le plus puissant. Un fichier de configuration est employé afin de facilement modifier la répartition des différents modules. Cette modularisation, combinée à d'autres techniques comme l'utilisation de différents niveaux de détails lors de la représentation du poste, permet d'obtenir une performance en simulation plus qu'acceptable.

Malgré l'implication de cinq personnes à temps plein et 12 mois de travail, l'ampleur de la tâche a fait en sorte que le projet ESOPE-RV n'est qu'au stade de prototype. Pour rapprocher ce produit d'une éventuelle commercialisation, certaines modifications devront être effectuées. Ces modifications visent essentiellement à améliorer la qualité

de la représentation 3D, à accroître la sensation d'immersion ainsi qu'à augmenter la flexibilité du simulateur ESOPE-RV.

La qualité de la représentation 3D peut facilement être améliorée en utilisant un poste de travail plus puissante. Un tel poste de travail, par exemple un poste de travail de marque SGI, possède une carte graphique supportant l'application de textures sur des polygones qui permet la création d'un poste électrique beaucoup plus réaliste. Par exemple, les transformateurs pourraient être verts avec un aspect métallique plutôt que d'être simplement vert mat et le sol dans le poste pourrait être fait de petites roches plutôt que d'être gris mat. La vitesse de la simulation serait également améliorée par l'utilisation d'une station plus puissante et plus récente comme la station O2 dont le prix d'achat est d'environ 8000 dollars (US). Cependant, l'emploi de polygones texturés avec un nouveau poste de travail nécessite quelques modifications mineures au logiciel ESOPE-RV. Un autre aspect pouvant être amélioré est la sensation d'immersion dans l'environnement virtuel. Le moyen préconisé est l'utilisation d'un casque ou de lunettes avec écran intégré. En effet, ces périphériques projettent devant les yeux l'environnement virtuel alors que pour la combinaison lunettes/écran utilisée jusqu'à maintenant, l'environnement virtuel est présenté sur un écran d'ordinateur et les lunettes ne font qu'ajouter un effet de profondeur à l'environnement virtuel. Plusieurs entreprises fabriquent des casques ou des lunettes à écrans intégrés qui projettent l'image devant les yeux et dont le prix d'achat est beaucoup plus abordable. Par exemple, la compagnie Virtual I-O de Seattle vend une lunette autour de 800

dollars (US). Ces lunettes sont supportées par l'outil de conception d'environnement virtuel, WorldToolKit de Sense8, et ne requièrent donc que très peu de modifications au code pour être intégrées au projet ESOPE-RV.

Un autre aspect du projet ESOPE-RV pouvant être amélioré est le système de reconnaissance vocale. Dans son état actuel, ce système est de type dépendant de l'utilisateur: chaque utilisateur doit « apprendre » au système à reconnaître sa voix. Il existe également des systèmes de reconnaissance vocale de type indépendant de l'utilisateur. Avec ce genre de système, l'utilisateur n'a pas à entraîner le système à reconnaître son timbre de voix. Il existe quelques systèmes qui sont en mesure de comprendre le français dont le système VoiceType d'IBM (IBM, 1997).

Le coût d'achat des stations de travail et de l'outil de création de l'environnement virtuel est assez substantiel (plus de 10 000\$ par station SGI en 1997 et 5 000\$ pour WorldToolkit) et pourrait diminuer l'accessibilité du simulateur ESOPE-RV. Une alternative serait d'utiliser des ordinateurs de type PC avec la version de WorldToolkit pour Windows NT. Le prix de WorldToolkit est le même mais le prix d'un ordinateur de type PC haut de gamme est de moins de 5 000\$. Une seconde alternative encore moins coûteuse consiste à utiliser un outil de création de l'environnement virtuel gratuit comme Mirage (Maurel, 1997). Cet outil peut être exécuté sur un ordinateur de type PC équipé du système d'exploitation Linux, qui est une variante d'UNIX

disponible gratuitement. Cette dernière approche nécessiterait cependant une réécriture complète du simulateur ESOPE-RV.

Les améliorations proposées jusqu'à maintenant concernent uniquement le matériel informatique et les périphériques. Le coeur du simulateur ESOPE-RV, soit les différents processus, pourrait également subir des améliorations afin de le rendre plus flexible et plus complet. Par exemple, le simulateur dans son état actuel permet d'effectuer la conversion des schémas unifilaires d'une seule configuration de barres de poste: *double-bus double-breaker*. La procédure de conversion pourrait être modifiée afin de supporter les autres configurations fréquentes afin de créer des postes virtuels dans les autres configurations de barres.

Un autre élément pouvant être amélioré est l'intégration du valideur de manœuvres au projet ESOPE-RV. En effet, un nombre limité de manœuvres réalisées avec ESOPE-RV est actuellement validé par le valideur de manœuvres. Les vérifications qu'il est possible de faire sur les appareils et sur l'installation sont également limitées aux plus fréquentes. De plus, il demeure impossible d'exécuter certaines opérations à partir d'ESOPE-RV. Par exemple, l'ouverture d'un nouveau schéma unifilaire n'est possible qu'à partir de l'interface traditionnelle du valideur de manœuvres.

L'amélioration représentant le plus grand défi est sans doute la création automatique de la salle de commande du poste. En effet, la version actuelle de la salle de

commande du schéma type d'ESOPÉ-RV a été dessinée manuellement avec un logiciel de dessin 3D. Une salle de commande créée de façon automatique ajouterait beaucoup plus de flexibilité au simulateur 3D. Cette automatisation pose cependant certaines difficultés au niveau du positionnement des manettes, des boutons, des cadrans de lecture et des identificateurs d'appareils. L'ampleur de ce projet permettrait certainement d'en faire un sujet de maîtrise.

Bibliographie

AKIYOSHI, M., MIWA, S., UEDA, T., NISHIDA, S. (1995). Learning environment for maintenance of power equipment using virtual reality, Proceedings of the 5th International Conference on Image Processing and its Applications, IEE Conference Publication n 410 p 331-335

APPLE COMPUTERS INC. (1991) Audio Interchange File Format AIFF-C: A revision to include compressed audio data, Draft version.

AT&T (1990). UNIX System V Release 4 Programmer's Guide, Prentice-Hall, Englewood Cliffs, NJ, USA

AUDET, J., STARSBOURG, C. (1993) Support pour formation de plan de manœuvres GCR-1362, document Hydro-Québec, service Automatismes, Télécommunications & Formation technique, division Formation technique - Réseau.

AUXILIUM TECHNOLOGIES INC. (1997). Manuel d'évaluation du valideur de manœuvres, Auxilium Technologies inc, Longueuil, Québec, Canada,

BASSLER E. (1994). Programming on Silicon Graphics System : an Overview, Silicon Graphics inc., document 007-2476-001

BEARE, ARTHUR N. (1997). Field Operation Power Switching Safety: An overview of the research study, Power Switching safety and Realibility Seminar, Columbia, Maryland, USA, 9-10 juin 1997

BERGER, JOSEPH E., DINH, LOAN T., MASIELLO, MICHAEL F., SCHELL, JESSE N. (1994). NVR: A system for networked virtual reality, Proceedings of the

International Conference on Multimedia Computing and Systems (Boston, MA, USA),
p 37-44.

BREEN, PAUL T. JR., SCOTT, WALTER G. (1995). Virtual reality applications in T&D engineering, Proceedings of the 1995 39th Annual Conference on Rural Electric Power Conference (Nashville, TN, USA), p B5-1/-6.

BURD, TOM (1997). The CPU InfoCenter, <http://infopad.eecs.berkeley.edu/cic>, U.C. Berkeley

CATER, JOHN P. (1994). Approximating the senses smell/taste: odors in virtual reality, Proceedings of the 1994 IEEE International Conference on Systems, Man and Cybernetics. Part 2 (of 3) (San Antonio, TX, USA), 94CH3571-5, p 1781.

CAUDELL, THOMAS P. (1995). Introduction to augmented and virtual reality, Telemanipulator and Telepresence Technologies, , Proceedings of SPIE - The International Society for Optical Engineering (Boston, MA, USA), p 272-281, v 2351

COMER, DOUGLAS E. (1991). Interconnecting with TCP/IP, volume 1, second edition, Prentice Hall, Englewood Cliffs, NJ, 07632, USA

DAVID, G. (1996). Operator training system moves toward virtual reality, Utility Automation, numéro de septembre/octobre, p. 28-29.

DEFANTI, THOMAS A., BROWN, MAXINE D., STEVENS, RICK (juillet 1996). Virtual reality over high-speed networks, IEEE Computer Graphics and Applications v 16 n 4. p 42-43

DORF C. RICHARD (1993). The Electrical Engineering Handbook, CRC Press, Boca Raton, Florida, USA

DUMAY, A.C.M.(1995). Triage simulation in a virtual environment, Interactive Technology and the New Paradigm for Healthcare. Medicine meets Virtual Reality III Proceedings, 19-22 Jan. 1995, San Diego, CA, USA

DUPONT, PIERRE (juin 1996). Virtual reality today, Computer Bulletin, v 8 n pt 3, 1996, p 14-18

FISHER, SCOTT S. (Ed.), MERRITT, JOHN O. (Ed.), BOLAS, MARK T. (Ed.) (1996). Stereoscopic Displays and Virtual Reality Systems III, Proceedings of SPIE - The International Society for Optical Engineering (San Jose, CA, USA) v 2653 1996. p 342

GARANT E., DAIGLE A., DESBIENS P., RIZZI J.-C., OKAPUU-VON VEH A., SHAIKH A., GAUTHIER R., MALOWANY A.S. ,MARCEAU R.J., (Juin 1995). Three-dimensionnal Modeling of a Virtuel Reality Operator Training Simulator, Stockholm Power Tech International Symposium on Electric Power Engineering, 18-22 Juin 1995, Stockholm, Suède, p. 31-6 vol. 4.

GARANT E., DESBIENS P., DAIGLE A., RIZZI J.-C., OKAPUU-VON VEH A., SHAIKH A., GAUTHIER R., MARCEAU R.J., MALOWANY A.S. (Mai 1995) A virtual reality training system for Power-Utility Personnel, IEEE PACRIM95 Conférence, 17-20 Mai, 1995, Victoria, BC, Canada.

GIGANTE (1993). Virtual Reality: Definitions, History and Applications, Virtual Reality System, chapitre 1 Academic press, San Diego, Ca,

GOTO, ATSUSHI, INOUE, RYUJI, TEZUKA, TETSUO, YOSHIKAWA, HIDEKAZU (1995). Research on tele-operation using virtual reality, Proceedings of the 1995 4th IEEE International Workshop on Robot and Human Communication (Tokyo, Japon), 95TH8115, p 147-152

GREENHALGH, CHRIS, BENFORD, STEVE (1995). MASSIVE: a distributed virtual reality system incorporating spatial trading, Proceedings of the 15th International Conference on Distributed Computing Systems (Vancouver, Canada), p 27-34

HAMILTON, PAUL, CHEN, SHIPING, HINTZ, TOM (1996). Commercial virtual reality system based on parallel and distributed, Proceedings of the 1996 2nd International Symposium on Parallel Architectures, Algorithms, and Networks (Beijing, Chine), 96TB100044. p 454-459

HARTSHORN, MICHAEL J., HERZYK, PAWEL, HUBBARD, RODERICK E. (mars 1995). Exploring molecular structure by virtual reality, Trends in Biotechnology v 13 n 3, p 83-85

HODGES, LARRY F., KOOPER, ROB, MEYER, THOMAS C. (juillet 1995). Virtual environments for treating the fear of heights, Computer v 28, p 27-34

HORSPOOL, NIGEL R. (1992). The Berkeley UNIX Environnement, second edition, Prentice-Hall Canada, Scarborough, On, Canada

HUNTER ET AL. (1993). A teleoperated Microsurgical Robot ans Associated Virtual Environnement for Eye Surgery, Presence, vol. 2, no. 4. pp. 256-280,

IBM (1997). La famille des Systèmes de Reconnaissance Vocale IBM,
<http://www.ibm.fr/france/vtd/prodgold.htm>

ISDALE, JERRY (1993). What is Virtual Reality ? A homebrew Introduction, article USENET

JOHNSON, R. COLIN (1995). VR : Hazardous to your health? – Scientists sound safety alarm over virtual reality, Electronic Engineering Times, June 5 1995, page 1.

KALAWSKY, ROY S (1993). The science of virtual reality and virtual environments, Addison-Wesley Publishing, Wokingham, England,

KUHL, JON, EVANS, DOUGLAS, PAPELIS, YIANNIS (juillet 1995). The Iowa Driving Simulator: an immersive research environment, Computer v 28, p 35-41

LIRVALL, PETER (Octobre 1995). Simulated maintenance a virtual reality, Nuclear News v 38, p 32-4

LIU, WENWEI, LI, JINTAO (1996). New approach to collision detection in virtual reality, Fourth Int. Conference on Computer-Aided Design and Computer Graphics (Wuhan, Chine), p 197-202

LOFTIN, R. BOWEN, KENNEY, PATRICK J. (septembre 1995). Training the Hubble Space Telescope flight team, IEEE Computer Graphics and Applications v 15, p 31-37

LOFTIN, R.B.; KENNEY, P. (1995). Training the Hubble space telescope flight team, IEEE Computer Graphics and Applications, Vol: 15 Iss: 5 p. 31-7, Sept., USA

MAUREL, C. (1997). Mirage : un nouvel outil de développement pour la réalité virtuelle sous X Windows, Mémoire Sc. App, École Polytechnique de Montréal.

MCLIN, D.M., (1995). Combining virtual reality and multimedia techniques for effective maintenance training, Proceedings of the SPIE - The International Society for Optical Engineering, 11-13 Oct. 1995, Washington, DC, USA

MCLIN, DAVID M., CHUNG, JAMES C. (1996). Combining virtual reality and multimedia techniques for effective maintenance, 24th AIPR Workshop on Tools and Techniques for Modeling and Simulation (Washington, DC, USA), v2645 p 204-210

MIDDLETON, VICTOR E., MCINTYRE, ROBERT T. III, O'KEEFE, JOHN IV (1993). Virtual reality and analytical simulation of the soldier, Proceedings of the Winter Simulation Conference (Los Angeles, CA, USA), p 1048-1052

MINER, NADINE E. (1994). Interactive virtual reality simulation system for robot control and operator training, Proceedings of the 1994 IEEE International Conference on Robotics and Automation (San Diego, CA, USA), p 1428-1435

MIZUKAMI, YUICHI, MORITA, SHIGERU, ASANO, KUNIHIRO, KAMIJI, NOBORU (décembre 1995). Substation simulation system using virtual reality technology, Electrical Engineering in Japan (English translation of Denki Gakkai Ronbunshi) v 115 n 8 15, p 109-120

MR TOOLKIT (1996). The MR Toolkit Virtual Reality Software, Edmonton, Al, Canada

OKAPUU-VON VEH, A. MARCEAU R.J., MALOWANY A.S., DESBIENS P., DAIGLE A., GARANT E., GAUTHIER R., SHAIKH A., RIZZI J.-C.(1996) Design

and operation of a virtual reality operator-training system, IEEE Transactions on Power Systems Vol: 11 Iss: 3 p. 1585-91, Août 1996.

PAPE, DAVE (juillet 1996). Hardware-independent virtual reality development system, IEEE Computer Graphics and Applications v 16 n 4, p 44-47

PAPKA, MICHAEL E., STEVENS, RICK (1996). UbiWorld: an environment integrating virtual reality, supercomputing and design, Proceedings of the 1996 5th IEEE International Symposium on High Performance Distributed Computing, Syracuse, NY, USA, 96TB100069. p 306-307

PELI, ELI (Juillet 1995). Real vision & virtual reality, Optics and Photonics News v 6 n 7. 6 pages

POLIS, MICHAEL F., GIFFORD, STEPHEN J., MCKEOWN, DAVID M.JR. (juillet 1995). Automating the construction of large-scale virtual worlds, Computer v 28, p 57-65

POPE, STEPHEN TRAVIS, FAHLEN, LENNART E. (1993). Use of 3-D audio in a synthetic environment: an aural renderer for a distributed virtual reality system, IEEE Annual Virtual Reality International Symposium (Seattle, WA, USA), p 176-182

POUNTAIN, DICK (juillet 1996). VR meets reality, Byte v 21 p 93-4+ 1996

QUINN, ROBERT, MCGUIRE, AUSTIN (1996). Implementation and assessment of a low cost shipboard bridge team training simulator using virtual reality technology, Proceedings of the 1996 National Technical Meeting (Santa Monica, CA, USA), p 407-411

ROBB, RICHARD A., HANSON, DENNIS P., CAMP, JON J. (janvier 1996).
Computer-aided surgery planning and rehearsal at Mayo Clinic, Computer v 29 p 39-47

ROEHL, BERNIE (1995). Distributed virtual reality - an overview, Proceedings of the 1995 1st Annual Symposium on the Virtual Reality Modeling Language (San Diego, CA, USA), p 39-43

ROSENBLUM, LAWRENCE, DURBIN, JIM, OBEYSEKARE, UPUL (novembre 1996). Shipboard VR: from damage control to design, IEEE Computer Graphics and Applications v 16 p 10-13

SADEGHI, TOM; WALL, SEAN (1995). Virtual reality tool for flight control law verification, Proceedings of the IEEE 1995 National Aerospace and Electronics Conference, 95CH35797. p 817-824

SENSE8 (1997). WorldToolKit for Silicon Graphics Home Page,
<http://www.sense8.com/products/wtksgi.html>, Mill Valley, Ca, USA

SHAIKH A., GARANT E., OKAPUU-VON VEH, A., MALOWANY A.S., DAIGLE A., DESBIENS P., RIZZI J.-C., MARCEAU R.J., GAUTHIER R. (1995). Alternative manipulation strategies in a virtual reality training system, Congrès canadien en génie électrique et informatique 1995, Montréal, 5-8 sept., 1995

SILVERIO C.J. ET FERGUSON W. (1994). C Language Reference Manual, Silicon Graphics inc., document 007-0701-080

STANNEY, KAY (1995). Realizing the full potential of virtual reality: human factors issues that could stand in the way, Proceedings of the 1995 IEEE Annual Virtual Reality International Symposium (Triangle Park, NC, USA), 95CH35761 p 28-34

STANSFIELD, SHARON, MINER, NADINE, SHAWVER, DAN, ROGERS, DAVE (1995). Application of shared virtual reality to situational training, Proceedings of the 1995 IEEE Annual Virtual Reality International Symposium (Triangle Park, NC, USA) 95CH35761. p 156-161

STARK, LAWRENCE W. (1995). How virtual reality works: illusions of vision in "real" and virtual environments, Proceedings of SPIE - The International Society for Optical Engineering (San Jose, CA, USA), v 2411 p 277-287

STEVENS, W. RICHARD (1990). Unix Network Programming, Prentice-Hall, chapitre 3, p. 87-170.

STEVENS, W. RICHARD (1992). Advanced Programming in the Unix environnement, Adisson-Wesley Publishing Co., chapitre 12.

STRAUSS, STEPHEN (1995). Cybersickness: the side effects of virtual reality, Technology Review v 98 p 14-16 Juillet 1995

STYTZ, M.R.(1994). An overview of current virtual reality research and development projects by the United States Department of Defense, Proceedings of the Fourth Annual Conference on Virtual Reality p. 152-9, 31 Jan.-2 Feb. London, UK

TAM E.K., MAUREL C., DESBIENS P., MARCEAU R.J., MALOWANY A.S., GRANGEL L., (1997). A low-Cost, PC-Oriented Virtual Environment for Operator Training Simulators, Power Industry Computer Applications 97.

VALENTINO, DANIEL J., BHUSHAN, VIKAS M.D., KISS, RICHARD, HARRELD, MICHAEL, LUFKIN, ROBERT B., VINUELA, FERNANDO, GIBSON, DAVID M.D. (1994). Virtual reality in radiology: innovative applications of PACS in Neuroradiology, Proceedings of SPIE - The International Society for Optical Engineering (Newport Beach, CA, USA) v 2164, p 393-401

VERBEX VOICE SYSTEM (1990). Verbex 7000 Grammar Developpement Manuel, Edison, NJ, USA

WARE, COLLIN, ARTHUR, KEVIN, BOOTH, KELLOGG S. (1993). Fish tank virtual reality, Proceedings of the Conference on Human Factors in Computing Systems - INTERACT '93 and CHI '93 (Amsterdam, Pays-bas), p 37-42

WHELESS, GLEN H., LASCARA, CATHY M., VALLE-LEVINSON, ARNOLDO (juillet 1996). Virtual Chesapeake Bay: interacting with a coupled physical/biological model, IEEE Computer Graphics and Applications v 16 p 52-7

WHITWARD, LANE (septembre 1994). How real are the benefits of virtual reality? Design Engineering (London), p 41-45

WILEY, DOUGLAS J., DUCKWORTH, ALLEN N. (1996). Adaptive level of detail technique for real-time display in virtual reality, Proceedings of SPIE - The international Society for Optical Engineering (Washington, DC, USA) v 2645 p 192-203

WITTENBERG, G. (1995). Training with virtual reality, Assembly Automation Vol: 15 Iss: 3 p. 12-14, MCB University Press, UK.

WORLD, LINDA (septembre 1995). The reality of cybersickness, IEEE Computer Graphics and Applications v 15 p 95

WORLDTOOLKIT, voir SENSE8 (1997).

WU, JING-LONG, MORITA, SEIKI, KAWAMURA, SADA0 (1996). Human sensory fusion on visual and tactile sensing for virtual reality, Proceedings of the 1996 13th IEEE International Conference on Robotics and Automation. Part 3 (of 4) (Minneapolis, MN, USA), 96CB35857. p 2365-2370

YOUN, JI-HOON, WOHN, K. (1993). Realtime collision detection for virtual reality applications, 1993 IEEE Annual Virtual Reality International Symposium (Seattle, WA, USA), 93CH3336-5 p 415-421

Annexes

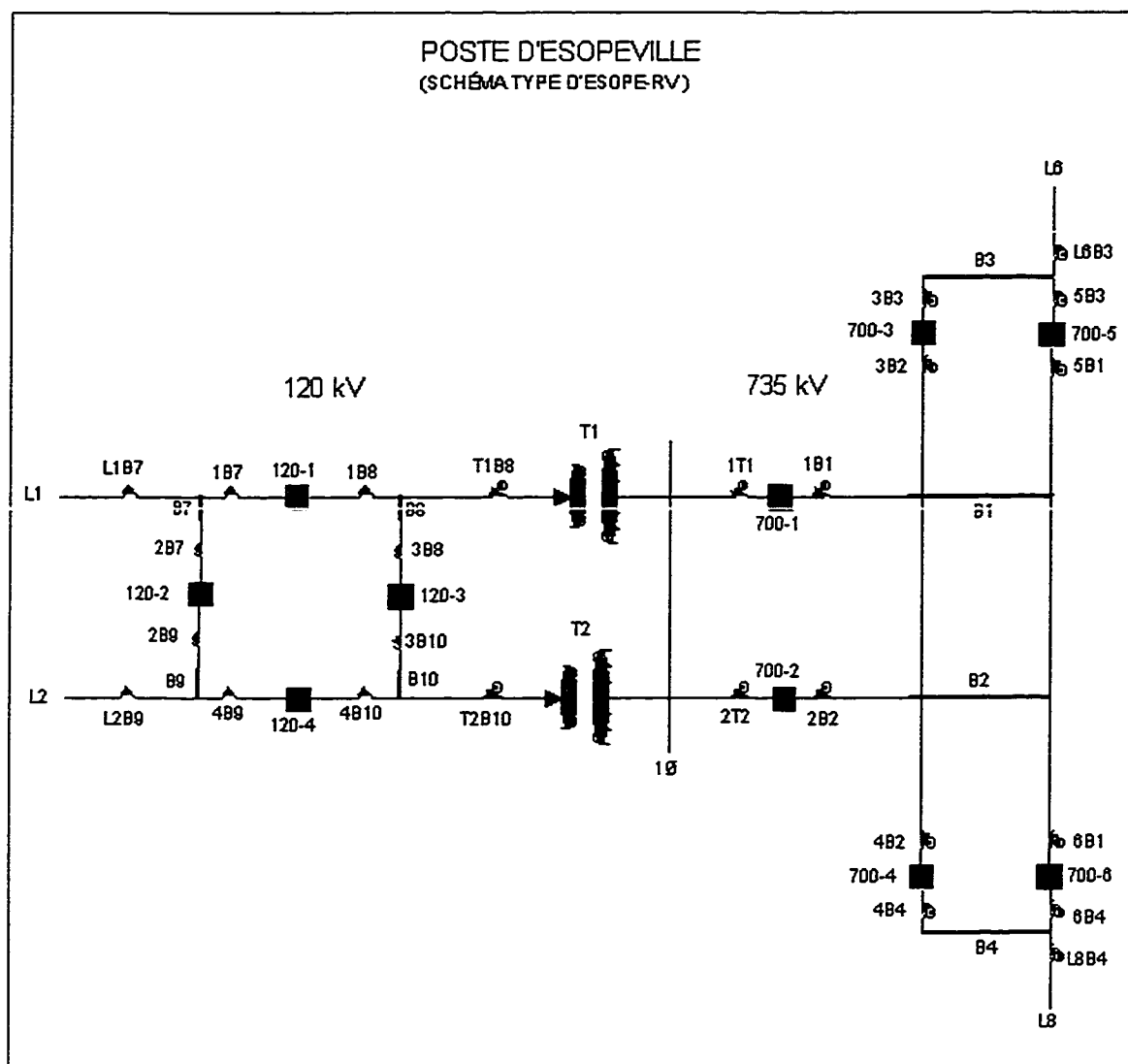
1. Plan de manœuvre type

BUT: Isoler le disjoncteur 700-3.

TRAVAIL: Ouverture du disjoncteur 700-3 et des sectionneurs 3B3 et 3B2.

1. Verification des conditions d'exploitations (salle de commande).
2. Mettre le mode tele du disjoncteur 700-3 a local (salle de commande).
3. Inspection de la zone concernee (inspecte tous les equipements a manœuvrer (equipement)).
4. Ouverture du disjoncteur 700-3 (salle de commande).
5. Verification(#1) au panneau de l'ouverture du disjoncteur (salle de commande).
6. Lecture du courant dans le disjoncteur (verification #2, salle de commande).
7. Mettre le mode tele du sectionneur 3B3 a local (salle de commande).
8. Ouverture du sectionneur 3B3 (equipement).
9. Verification(#1) a l'equipement de l'ouverture du sectionneur (equipement).
10. Verification(#2) au panneau de l'ouverture du sectionneur (salle de commande).
11. Mettre le mode tele du sectionneur 3B2 a local (salle de commande).
12. Ouverture du sectionneur 3B2 (equipement).
13. Verification(#1) a l'equipement de l'ouverture du sectionneur (equipement).
14. Verification(#2) au panneau de l'ouverture du sectionneur (salle de commande).
15. Remettre les appareils a mode tele (salle de commande).
16. Verification des conditions d'exploitations (salle de commande).

2. Schéma unifilaire type



3. Structures de données utilisées

```

struct message{
char *chaine; /* renferme le message */
CARACMESSAGE caracteristiques; /* caracteristiques du message */
char chainepourwinesope[64]; /* renferme la commande pour le
    Valideur */
char parachainepourwinesope[64]; /*renferme les parametres de
    la commande */
char chainerecherchee[256]; /* chaîne desserre dans la réponse
    de Win-Esope */
char resultataccumule[LONGEURMESSAGEMAX]; /*tampon pour
    messages condi */
char entetemessage[64]; /* entete du message */
int EffacerCeMessage; /* flag si on doit effacer le mess. */
int RetourneListe; /* flag si message retourne une liste */
int TraitementTermine; /* flag si traitement du mess. est
    fini */
int EnTraitement; /* flag si message presentement en
    traitement */
int NumeroDeFonction; /* numero de la fonction du Valideur
    appelee*/
int Accepte; /* Accepte = 1 si manœuvre est accepte */
int CodeErreur; /* code de l'erreur si erreur */
int EnvoyerWinEsope; /* message destine pour winesope?*/
MESSAGE *messagesuivant; /* message suivant dans la file */
MESSAGE *messageprecedent; /* message précédant dans la file */
} MESSAGE ;

typedef struct {
    char *origine; /* emetteur, ESO, VBX, ERV, A_V */
    char *dest; /* destinataire, ESO, VBX, ERV, A_V */
    char classe; /* type de message, NAV, INT, ERR, MAN */
    char operation; /* operation a effectuer */
    char element; /* element traite */

```

```
char condition; /* condition , si presente */
char parampanneau; /* parametre */
char origine_mess; /* emetteur du message */
int socket_retour; /* socket utilise pour resultat */
int NumeroMessage; /* numero sequentiel */
} CARACMESSAGE;
```